Revisiting Radius, Diameter, and all Eccentricity Computation in Graphs through Certificates

Feodor Dragan, Michel Habib, Laurent Viennot

Revisiting Radius, Diameter, and all Eccentricity Computation in Graphs through Certificates*

Feodor Dragan[†] Michel Habib[‡] Laurent Viennot[§]

March 13, 2018

Abstract

We introduce notions of certificates allowing to bound eccentricities in a graph. In particular, we revisit radius (minimum eccentricity) and diameter (maximum eccentricity) computation and explain the efficiency of practical radius and diameter algorithms by the existence of small certificates for radius and diameter plus few additional properties. We show how such computation is related to covering a graph with certain balls or complementary of balls. We introduce several new algorithmic techniques related to eccentricity computation and propose algorithms for radius, diameter and all eccentricities with theoretical guarantees with respect to certain graph parameters. This is complemented by experimental results on various real-world graphs showing that these parameters appear to be low in practice. We also obtain refined results in the case where the input graph has low doubling dimension, has low hyperbolicity, or is chordal.

1 Introduction

The radius and diameter of a graph are part of the basic global parameters that allow to apprehend the structure of a practical graph. More broadly, the eccentricity of each node, defined as the furthest distance from the node, is also of interest as a classical centrality measure [19]. It is tightly related to radius which is the minimum eccentricity and to diameter which is the maximum eccentricity. On the one hand, efficient computation of such parameters is still theoretically challenging as truly sub-quadratic algorithms would improve the state of the art for other "hard in P" related problems such as finding two orthogonal vectors in a set of vectors or testing if one set in a collection is a hitting set for another collection [2]. A sub-quadratic diameter algorithm would also refute the strong exponential time hypothesis (SETH) [25] and would improve the state of the art of SAT solvers as noted for similar problems in [24]. On the other hand, a line of practical algorithms has been proposed based on performing selected Breadth First Search traversals (BFS) [23, 27, 28, 8, 5] allowing to compute the diameter of very large graphs [3]. However, such practical efficiency is still not well understood.

What are the structural properties that make practical graphs tractable? This paper answers this question with the lens of certificate, that is a piece of information about a graph that allows to compute its radius and diameter in truly sub-quadratic time. We propose a notion of certificate tightly related to the class of algorithms based on one-to-all distance computations from selected nodes. Existing practical algorithms fall into this category that we call one-to-all distance based algorithms. Based on this approach, we propose algorithms with proven guarantees with respect to several graph properties which appear to be generally met in practice.

Another intriguing question concerns the relationship between diameter and radius computations. The most advanced algorithms [28, 5] compute both parameters at the same time. Would

*Work supported by ANR Distancia.

[†]Kent State University
[‡]Paris Diderot University
[§]Inria

computing one parameter help for computing the other? We answer by the affirmative based on the notion of certificate.

The paper is presented in the context of unweighted undirected graphs but all the notions and algorithms extend to the weighted and/or directed cases.

1.1 Our contribution

We introduce the notion of certificate as a set of nodes such that the distances from these nodes to all nodes (rather than all-to-all pairs) allow to deduce the value of the radius or the diameter with certainty. Given a graph G with radius r, we define a *radius certificate* as a set L of nodes such that any node of G is at distance at least r from a node of L. Given in addition a node c with eccentricity r, the set L allows to certify that the radius $\mathrm{rad}(G)$ of G, i.e. the minimum eccentricity, is indeed r: we can compute r with a BFS from c and certify that all nodes have eccentricity r or more by checking that their distance to some node in L is at least r using $|L|$ BFS traversals. If L has size $o(n)$, this opens the possibility of breaking the quadratic barrier for radius computation if one can efficiently find a small certificate when there exists one. This raises the problem of approximating the minimum certificate for radius. Interestingly, the size R of the minimum radius certificate gives a lower bound on the complexity of one-to-all distance based algorithms for radius: such an algorithm must perform at least $R/2$ one-to-all distance computations. We also raise similar approximation problems for diameter and all eccentricity computations.

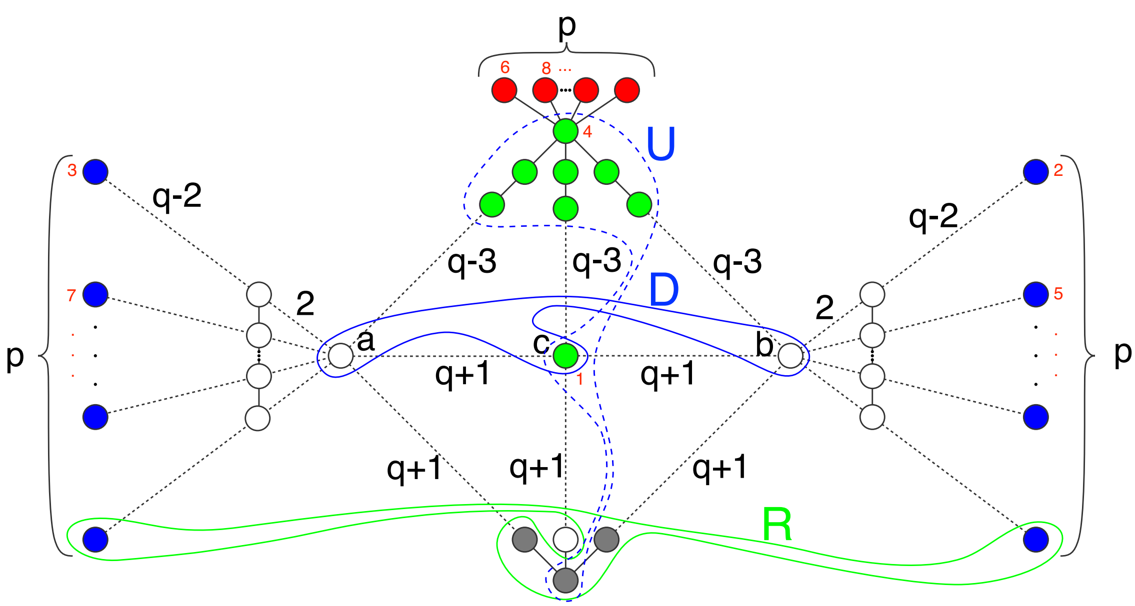

Figure 1: An example of graph $BT_{p,q}$ (for $p \geq 2$ and $q \geq 6$) with small certificates for radius (R), diameter (D) and all eccentricities (R, U). Its diameter is $4q-2$ (eccentricity of blue nodes). Its radius is $2q+1$ (eccentricity of green nodes). Plain lines correspond to edges while a dashed line with label ℓ corresponds to a path of length ℓ.

We show that a radius certificate can be formally defined as a covering of the node set with complementary of open balls of radius $\mathrm{rad}(G)$ (excluding nodes at distance $\mathrm{rad}(G)$). We also define a diameter certificate as a covering with balls $B[x, \mathrm{diam}(G) - e(x)]$ of radius $\mathrm{diam}(G) - e(x)$ where $\mathrm{diam}(G)$ is the diameter of the graph and $e(x)$ is the eccentricity of the center x of the ball. Similarly, an all eccentricity certificate can be defined by combining two coverings as a pair of lower/upper (see definitions in Section 3). Finding a minimum radius (or diameter) certificate is shown to be equivalent to minimum set cover. It is thus NP-hard while $O(\log n)$-approximation (only) is doable in polynomial time. Compared to set cover, it has an additional difficulty: the sets are not directly available and computing all of them would require quadratic time at least. It should be noted that these notions of certificate are independent of any algorithm: it is a graph property to have small or big certificates. As an example, for odd k, a $k \times k$ square grid has a one-node diameter certificate (its center) and a radius certificate with four nodes (its corners).

Figure 1 presents an example of bow-tie shaped graph $BT_{p,q}$ for integral parameters $p \geq 2$ and $q \geq 6$ that has small certificates. The diameter certificate D contains three nodes a, b, c. The central (green) node c has eccentricity $2q+1$. Note that its eccentricity is minimal ($\mathrm{rad}(BT_{p,q}) = 2q+1$) and c is called a center. Any node at distance $r \leq \mathrm{diam}(BT_{p,q}) - (2q+1) = 2q-3$ from c has eccentricity at most $r + 2q + 1 \leq \mathrm{diam}(BT_{p,q})$ as it can reach any node v by following a path of length r to c and then a path from c to v of length $e(c) = 2q+1$ at most. In set-cover terms, c covers the ball $B[c, 2q-3]$. The rest of the graph is covered by the balls of radius q centered at a and b, implying that $D = \{a, b, c\}$ is a diameter certificate. The radius certificate R has five nodes such that any node is at distance $\mathrm{rad}(BT_{p,q}) = 2q+1$ at least from one of them. In other words, the complement of open balls of radius $\mathrm{rad}(BT_{p,q})$ centered on them cover the whole graph.

We propose algorithms for radius, diameter and all-eccentricity certificate computation (as a byproduct, our algorithms also provide radius, diameter and all eccentricities). They follow a primal-dual approach that allows to obtain guarantees on the size of computed certificates and on the number of BFS traversals performed with respect to graph parameters that seem to be low in practical graphs. Our experiments on practical graphs from various sources show that these graphs not only have small certificates but also small coverings with much reduced sets: we can still cover the node set with few complementary of balls with increased radii (resp. decreased radii) compared to radii required for a radius (resp. diameter) certificate. Such properties explain the efficiency of a primal dual approach. Although our algorithms have some similarities with previous algorithms, this primal-dual flavor was not noticed before. They have similar performances in practice but provide significantly smaller certificates. Their proven guarantees also make them more robust. In particular, our radius and diameter algorithms handle the graph $BT_{p,q}$ of Figure 1 with $O(1)$ BFS traversals while previous exact algorithms require $\Omega(p)$ BFS traversals.

Our experiments show a striking phenomenon concerning specifically lower-bounding eccentricities (as in radius computation) that we call "*antipode sparsity*". Given a ranking of the nodes (e.g., their ID order), we define the *antipode* of a node u as the node at furthest distance from u having highest rank (the ranking is used for breaking ties among nodes at the same distance). We say that a node is an antipode if it is the antipode of some other node. We observe that practical graphs have very few antipodes for several rankings (i.e., large groups of nodes share the same antipode): most the practical graphs tested (with up to hundred of thousands of nodes) have less than 100 antipodes. Although our notion of antipode is reminiscent of the usage of antipodes on the Earth, we see that it can significantly deviate from it. On the sphere, the antipode of a point is the unique furthest point from it and the antipode of the antipode is the point itself. The same situation can be met in graphs such as a cycle or a grid torus. However it appears to be much different in practical graphs: the relation is highly asymmetric, most of the nodes have multiple furthest nodes (i.e., nodes at furthest distance from them) while there are very few antipodes overall. This situation is indeed highly favorable to one-to-all distance based algorithms as shown by the following theorem summarizing our algorithmic results.

Theorem 1 *Given a connected graph G having m edges and k antipodes overall (according to a given ranking), it is possible to compute:*

- *its radius, a center and a radius certificate of size k at most,*
- *its diameter, a diametral node and a diameter certificate of size $\pi_{1/3}$ at most where $\pi_{1/3}$ is the maximum packing size for open balls $B(u, \frac{1}{3}(\mathrm{diam}(G) - e(u)))$,*
- *all eccentricities, a lower certificate of size k at most and a minimum upper certificate U_{OPT},*

using $O(1)$ BFS traversals per node of associated certificates (i.e., in $O(km)$, $O(\pi_{1/3}m)$ and $O(km + |U_{OPT}|\, m)$ time respectively).

Concerning diameter (second item), we analyse a minimalist algorithm inspired by previous practical algorithms. A basic primal-dual argument implies that the maximum size π_1 of a packing for (closed) balls $B[u, \alpha(\operatorname{diam}(G) - e(u))]$ for $\alpha = 1$ is a lower bound of the minimum size of a diameter certificate. The above theorem thus indeed proves that this basic approach approximates minimum diameter certificate within a ratio of $\pi_{1/3}/\pi_1$. While the value $\pi_{0.8}$ (with radii reduced by a factor 0.8) appears to be generally small in practice, the $\pi_{1/3}$ bound can be much higher than π_1. However, this provides a first answer for the efficiency of practical diameter algorithms that can be complemented with the following observation. A second property often met by practical graphs is a high diameter to radius ratio $\operatorname{diam}(G)/\operatorname{rad}(G)$ (over 1.5 in our experiments) so that a large part of the graph is included in any ball $B[c, \operatorname{diam}(G) - \operatorname{rad}(G)]$ centered at a central node c. Such a ball corresponds to the nodes covered by adding c to a diameter certificate in the associated set cover problem. We confirm this with a refinement of the parameter $\pi_{1/3}$ in the above theorem when combining radius and diameter computation where a center c of the graph is used to initialize the basic diameter algorithm. We observe values for that refined parameter that are generally within a small constant factor of π_1. This graph property associated with high diameter to radius ratio and the discovery of a node with small eccentricity as part of diameter computation is thus our second element for explaining the efficiency of practical diameter algorithms.

Concerning radius computation, practical algorithms tend to perform even faster than predicted by the first point of the above theorem (including the algorithm analyzed in the theorem). This large diameter to radius ratio also allows us to give an intuition for this. Our radius algorithm iteratively selects a node with minimal eccentricity lower-bound (according to the radius certificate computed so far) and adds its antipode to the candidate radius certificate. We can show that the selected node is always in the intersection of all balls of radius $\operatorname{rad}(G)$ centered at previous discovered antipodes. As antipodes tend to have high eccentricity to graph-radius ratio (in the order of the diameter to radius ratio), this intersection quickly shrinks toward the set of graph centers. As selecting a node with minimal lower-bound combined with discovering high eccentricity nodes is a classical approach, this gives a second element for understanding the efficiency of practical radius algorithms. Note that the idea of using antipodes systematically for finding high eccentricity nodes is new.

Although we reuse classical algorithmic tools, our radius and all eccentricity algorithms rely on a new algorithmic technique that we call *minimum eccentricity selection* which has its own interest. It specifically leverages on antipode sparsity for enabling efficient selection of a node with minimum eccentricity within a set maintained online. Its amortized complexity is low when the number of antipodes is small. Interestingly, this technique also allows to design algorithms based on an oracle giving access to all eccentricities. Such algorithm can then be efficiently implemented using our technique as long as eccentricity values are used to iteratively select a node u such that $f(u, e(u))$ is minimal for a given computable function f satisfying some non-decreasing property. It is based on the idea of using antipodes to enhance a (lower) certificate until an adequate node is found. The technique also appears to be useful for optimizing diameter computation. We also introduce a new technique for diameter computation that we call *delegate certificate* in order to obtain both theoretical guarantees and efficient practical performances.

Finally, a surprising fact concerns the complexity of finding an optimum upper certificate (a certificate with minimum size that provides a tight upper-bound of the eccentricity of each node) as provided by our all eccentricity algorithm. Contrarily to radius and diameter certificates (as discussed above), it appears to be tractable in polynomial time. In comparison, finding an optimum lower certificate is also shown to be as hard as set-cover. Moreover, our all eccentricity algorithm roughly performs one BFS traversal per node of the optimum upper certificate when the number of antipodes is much smaller than the size of this upper certificate (as observed in our experiments). Note that this is close to the best possible for an algorithm based on one-to-all distance computations.

We additionally refine the performance analysis of our algorithms in particular cases when the input graph 1) has bounded doubling dimension, 2) has small hyperbolicity, or 3) is a chordal graph.

We believe that our certificate approach provides new insight on the efficiency of practical algorithms for radius and diameter, allows to propose more robust practical algorithms with complexity guarantees, and significantly enhance the state of the art for all eccentricity computation. Moreover, the new techniques proposed here could enable new types of radius and diameter algorithms with parametrized complexity.

1.2 Related work

The concept of certificate is somehow implicit in the method introduced in [27, 28] that consists in maintaining lower and upper bounds on the eccentricity of each node. After each BFS traversal these bounds are improved based on distances from the source of the traversal. The sources used for the BFS traversals performed by the algorithm form what we call a certificate. Contrarily to this approach, we distinguish nodes used for improving lower bounds (the lower certificate) from those used for improving upper bounds (the upper certificate). Our definition of lower certificate uses a looser lower-bounding inequality because of this distinction. The main approach proposed for diameter computation [27] consists in alternating nodes with small lower bound and nodes with large upper bound as BFS sources. This can be seen as a mix of our basic diameter algorithm with a heuristic for finding nodes with small eccentricities.

The two-sweeps heuristic [23] performs only 2 traversals to provide a diameter estimate that appears to be tight in practice. The idea is to use the last visited node in the first traversal to start the second traversal. It thus introduces the idea of using what we call antipodes as tentative diametral nodes. The technique was first introduced for trees [20] where it happens to be exact. It was also shown to provide good approximation (up to a small constant) for chordal graphs and various graph classes [14].

A four-sweeps heuristic is proposed in [8] and complemented with an exact diameter algorithm called iFub. The four-sweep heuristic performs twice the two-sweep method, using a mid-point of the longest path found in the first round as the starting point of the second one. The idea is that mid-points of longest paths make good candidates for central nodes or at least nodes with small eccentricity. The iFub method additionally inspects furthest nodes from the best candidate center found with the four-sweep heuristic until exact value of the diameter can be inferred.

The exact-sum-sweep method computes (exactly) both radius and diameter while performing few BFS traversals in practice [5]. It integrates many techniques proposed in previous practical algorithm plus an heuristic based on sum of distances that boosts the discovery of nodes with large eccentricity in an initial phase. It also handles the directed case in a very general manner.

The structure of random power law graphs is analyzed in [6] and the efficiency of practical diameter and radius algorithms is discussed for that type of graphs. It is shown that random power law graphs satisfy similar properties as those we insist on. The main argument proposed for efficiency of practical algorithms resides in the fact that such graphs have few furthest nodes (that is nodes that appear to be furthest from some other node). However, we observe much less antipodes than furthest nodes in practice and some graphs do have a fairly high number of furthest nodes. Our work provides a finer parameter and allows to extend such explanation to other types of practical graphs such as road networks and grid like networks.

Packing and covering of hyperbolic graphs with balls is investigated in [9], although slightly different problems are considered. It would be interesting to derive similar results in hyperbolic graphs for the collections of balls (or complementary of balls) we consider here.

1.3 Structure of the paper

We introduce basic graph and set-cover terminology in Section 2. The notions of certificate for radius, diameter, and all eccentricities are given in Section 3. We show how such notion can be related to one-to-all distance based algorithms in Section 4. Section 5 is devoted to our radius algorithm. We introduce in Section 6 the technique of minimum eccentricity selection which is the core of this radius algorithm. We analyse a basic diameter algorithm and propose an optimization based on radius computation and minimum eccentricity selection in Section 7. Computation of all eccentricities is studied in Section 8. Theorem 1 is a consequence of the theorems proven in Sections 5, 7 and 8. We present some experimental results in Section 9 about the measurement on various practical graphs of the parameters involved in our theorems. Section 10 is devoted to graphs with low doubling dimension: a refined algorithm for diameter computation is proposed and our radius and diameter algorithms are analyzed in terms of radius and diameter approximation respectively. Section 11 refines the analysis of our radius algorithm in the case of graphs with low hyperbolicity. Finally, we study chordal graphs in Section 12: we show that centers form a diameter certificate while diametral nodes form a radius certificate, this allows to derive a linear time algorithm for computing all eccentricities of a bounded degree chordal graph.

2 Preliminaries

Given an undirected unweighted graph G we denote by V its set of nodes. Let $d(u, v)$ be the distance between two nodes u and v in G, that is the length of a shortest path from u to v. The *eccentricity* $e(u)$ of a node u is the maximum length of a shortest path from u, that is $e(u) = \max_{v \in V} d(u, v)$. The *furthest nodes* of u are the nodes v at furthest distance from u, i.e., $d(u, v) = e(u)$. Given a ranking r of the nodes, the *antipode* $\mathrm{Antipode}_r(u)$ of a node u for r is its furthest node with highest rank. Formally, $\mathrm{Antipode}_r(u) = \mathrm{argmax}_{v \in V}(d(u, v), r(v))$ where pairs are ordered lexicographically. A node is called a furthest node (resp. an antipode) if it is a furthest node (resp. an antipode) of some other node. Given a set $W \subseteq V$, we let $\mathrm{Antipode}_r(W) = \{\mathrm{Antipode}_r(u) : u \in W\}$ denote the set of antipodes from nodes in W. The *diameter* $\mathrm{diam}(G) = \max_{u \in V} e(u)$ of G is the maximum eccentricity in G and the *radius* $\mathrm{rad}(G) = \min_{u \in V} e(u)$ is the minimum eccentricity in G. A *diametral node* b is a node with maximum eccentricity ($e(b) = \mathrm{diam}(G)$). A *central node* c (or simply *center*) is a node with minimum eccentricity ($e(c) = \mathrm{rad}(G)$). We let $B[u, r] = \{v \in V \mid d(u, v) \leq r\}$ (resp. $B(u, r) = \{v \in V \mid d(u, v) < r\}$) denote the (closed) ball (resp. open ball) with radius r centered at a node u. Similarly, we define its *coball* of radius r as $\overline{B}(u, r) = \{v \in V \mid d(u, v) \geq r\}$, that is the complementary of $B(u, r)$.

We restrict ourselves to algorithms based on one-to-all distance queries: we suppose that an algorithm DistFrom for one-to-all distances is given (typically BFS or Dijkstra). It takes a graph G and a node u as input and returns distances from u. More precisely, $\mathrm{DistFrom}(G, u)$ returns a vector D such that $D(v) = d(u, v)$ for all $v \in V$. In particular, $e(u)$ can be obtained as the maximum value in the vector and the antipode of u as the index with highest rank were this value appears in D. We may measure the complexity of an algorithm by the number of one-to-all distance queries it performs when its cost mainly comes from these operations. A one-to-all distance based algorithm accesses the graph only through one-to-all distance queries and relies solely on distances known from queries, triangle inequality, and non-negativeness of distances for bounding unknown distances.

Given a collection $\mathcal{S}$ of subsets of V such that $\cup_{S \in \mathcal{S}} S = V$, a *covering* with $\mathcal{S}$ is a sub-collection $\mathcal{C} \subseteq \mathcal{S}$ of sets such that their union covers all V: $V \subseteq \cup_{S \in \mathcal{C}} S$. (A set $S \in \mathcal{S}$ is said to cover elements in S.) Recall that the set-cover problem consists in finding a covering of minimum size. We define a *packing* for $\mathcal{S}$ as a subset $P \subseteq V$ such that any set of $\mathcal{S}$ contains at most one element in P. The denomination comes from the fact that elements of P correspond to pairwise disjoint subsets of the dual collection $\mathcal{S}^* = \{\{S \in \mathcal{S} \mid u \in S\} : u \in V\}$. A *hitting set* for $\mathcal{S}$ is a set P that intersects all sets of $\mathcal{S}$. (Equivalently, a hitting set can be defined as a covering for $\mathcal{S}^*$ but it may be more convenient to consider a collection rather than its dual.) We let $\pi(\mathcal{S})$ denote the maximum size of a packing for $\mathcal{S}$, and $\kappa(\mathcal{S})$ denote the minimum size of a covering with $\mathcal{S}$. As a covering must cover each element of a packing with distinct sets, we obviously have $\pi(\mathcal{S}) \leq \kappa(\mathcal{S})$ (weak duality). We say that a collection $\mathcal{R}$ is *restricted* compared to $\mathcal{S}$ if there exists a one-to-one mapping f from $\mathcal{R}$ to $\mathcal{S}$ such that $S \subseteq f(S)$ for all sets $S \in \mathcal{R}$. Note that this mapping then turns any covering with $\mathcal{R}$ into a covering with $\mathcal{S}$ and we thus have $\kappa(\mathcal{S}) \leq \kappa(\mathcal{R})$. Similarly, a packing for $\mathcal{S}$ is also a packing for $\mathcal{R}$ and we have $\pi(\mathcal{S}) \leq \pi(\mathcal{R})$. In other words, restricting the sets of a collection to smaller subsets increases maximum packing size and minimum covering size.

3 Lower and upper certificates for eccentricities

Our notion of certificate is based on the fact that knowing all distances from a given node x allows to derive some bounds on the eccentricities of other nodes:

$$\forall u \in V,\ d(u,x) \leq e(u) \leq d(u,x) + e(x). \tag{1}$$

The first inequality derives directly from the eccentricity definition while the second one is a consequence of the triangle inequality. A possibly tighter lower-bound of $\max\{d(u,x), e(u) - d(u,x)\}$ could be used as in [27] but this optimization does not allow to reduce drastically certificate size (see Section 4).

We say that a set L (resp. U) of nodes is a *lower certificate* (resp. an *upper certificate*) of G when it is used to obtain lower bounds (resp. upper bounds) of eccentricities in G. Given the distances from a node u to all nodes in $L \cup U$ and the eccentricities of nodes in U, we have the following lower and upper bounds for the eccentricity of any node u (as a direct consequence of Inequation 1):

$$e_L(u) \leq e(u) \leq e^U(u), \quad \text{where } \begin{cases} e^U(u) = \min_{x \in U} d(u,x) + e(x) \\ e_L(u) = \max_{x \in L} d(u,x) \end{cases}$$

A lower (resp. upper) certificate L (resp. U) is said to be *tight* when $e_L(u) = e(u)$ (resp. $e^U(u) = e(u)$) for all $u \in V$. An *all-eccentricty certificate* is defined as a pair L, U of a tight lower certificate L and a tight upper certificate U.

Given a bound D and a node x, we have $d(u,x) + e(x) \leq D$ if and only if $u \in B[x, D - e(x)]$. Given an upper certificate U we thus have $e^U(u) \leq D$ if and only if $u \in \cup_{x \in U} B[x, D - e(x)]$. A *diameter certificate* is a set U such that $e^U(u) \leq \mathrm{diam}(G)$ for all $u \in V$. Equivalently it can be defined as a covering with $\{B[x, \mathrm{diam}(G) - e(x)] : x \in V\}$ using balls whose radius equals $\mathrm{diam}(G)$ minus eccentricity of the center (and identifying a ball with its center).

Similarly, given a lower certificate L and a bound R, we obviously have $e_L(u) \geq R$ for all nodes u whose coball $\overline{B}(u,R)$ intersects L (i.e., there exists a node in L at distance R at least from u). We thus define a *radius certificate* as a L such that $e_L(u) \geq \mathrm{rad}(G)$ for all $u \in V$ or equivalently as a hitting set L for the collection $\{\overline{B}(u, \mathrm{rad}(G)) : u \in V\}$ of coballs of radius $\mathrm{rad}(G)$. As $x \in \overline{B}(u, \mathrm{rad}(G))$ if and only if $u \in \overline{B}(x, \mathrm{rad}(G))$, the collection of coballs of radius $\mathrm{rad}(G)$ is its own dual, and a radius certificate L can equivalently be defined as a covering for this collection.

Note that a tight lower certificate can equivalently be defined as a hitting set for the collection $\{\overline{B}(u, e(u)) : u \in V\}$. Similarly, a tight upper certificate can equivalently be defined as a covering with the collection $\{\{u \in V \mid d(u,x) \leq e(u) - e(x)\} : x \in V\}$.

Examples. A path with $2k+1$ nodes has a radius certificate with two nodes (the two extremities) and a diameter certificate with one node (its mid-point). More generally, any graph G such that $\mathrm{diam}(G) = 2\,\mathrm{rad}(G)$ has a one node diameter certificate (a center). It can be shown that any tree has a radius certificate of two nodes (two well chosen leaves) while its centers (at most two nodes) form a diameter certificate. A square grid has a radius certificate with four nodes (the corners) while its centers (at most four nodes) form a diameter certificate. As an extreme example of graph with large certificates, consider a cycle C. Its only radius and diameter certificates are both the whole set of its nodes. More generally, the whole set of nodes is the only diameter certificate of any graph where all nodes have same eccentricity (when radius equals diameter).

Hardness of approximation. Similarly to [9], we note that set cover can easily be encoded with ball cover: given a collection $\mathcal{S}$ of subsets of V of an instance of the set-cover problem, consider the split graph where the sets S of $\mathcal{S}$ form a clique and the elements $x \in V$ form a stable set so that the nodes x and S are adjacent if and only if $x \in S$. Without loss of generality, we may assume that no subset of $\mathcal{S}$ equals V (otherwise, the problem is trivial), no set is empty (otherwise, we can remove it) and that there exists two elements such that no set contains both of them (if needed, we add a singleton $\{z\}$ to $\mathcal{S}$ where z is a new dummy element added to V). In this graph, sets and elements have eccentricity 2 and 3 respectively. Any minimum diameter certificate is a covering with balls of radius 1 or 0 (if centered on a set or an element). One can easily transform it into a covering with balls of radius 1 centered at nodes corresponding to sets (only). It then corresponds to an optimal solution of the original set-cover problem. Now consider the complementary graph which is also a split graph where elements form a clique while sets form a stable set and where x and S are adjacent if and only if $x \notin S$. Similarly, a minimum radius certificate for this complementary graph corresponds to a covering with coballs of radius 2 centered at sets and is also an optimal solution to the original set-cover problem. For that graph, finding a radius certificate is equivalent to finding a tight lower certificate. The hardness of set-cover approximation [16] thus implies that computing a minimum diameter (resp. radius or tight lower) certificate is NP-hard and that no polynomial time algorithm can approximate it with a factor $(1 - o(1)) \log n$ unless $P = NP$. Surprisingly, we will see that finding a minimum tight upper certificate can be done in polynomial time.

4 Lower bound for radius computation

We first show that the notion of radius certificate is related to the minimum number of queries a one-to-all distance based algorithm must perform.

Theorem 2 *Given a graph G, if a one-to-all distance based algorithm for radius queries a set L of nodes then $L \cup \mathrm{Antipode}_r(L)$ is a radius certificate for any ranking r. Such a radius algorithm thus requires at least $\frac{1}{2}|L_{OPT}|$ one-to-all distance queries where $|L_{OPT}| = \kappa(\{\overline{B}(u, \mathrm{rad}(G)) : u \in V\})$ is the minimum size of a radius certificate.*

Proof. Consider a one-to-all distance based algorithm and let L denote the set of nodes it queries for one-to-all distances. A proof of correctness of the algorithm allows to conclude that all nodes have eccentricity $\mathrm{rad}(G)$ at least based on triangle inequality and the distances known to the algorithm. That is for each node u, there is a node v such that we can prove $d(u, v) \geq \mathrm{rad}(G)$ based on triangle inequality and distances from nodes in L. Consider a node v such that the proof uses a minimum number of triangle inequalities. If neither u nor v is in L, the proof must use a triangle inequality $d(u, v) \geq |d(u, x_1) - d(v, x_1)|$ for some node x_1 and a proof of $|d(u, x_1) - d(v, x_1)| \geq \mathrm{rad}(G)$. In the case $|d(u, x_1) - d(v, x_1)| = d(u, x_1) - d(v, x_1)$, we would have a shorter proof $d(u, x_1) \geq \mathrm{rad}(G)$ in contradiction with the choice of v. We thus consider only the case $|d(u, x_1) - d(v, x_1)| = d(v, x_1) - d(u, x_1)$. We then have a proof of $d(v, x_1) \geq \mathrm{rad}(G) + d(u, x_1)$. Either $x_1 \in L$ or the proof uses a node x_2 such that $|d(v, x_2) - d(x_2, x_1)| \geq \mathrm{rad}(G) + d(u, x_1)$. The choice of v again implies $|d(v, x_2) - d(x_2, x_1)| = d(v, x_2) - d(x_2, x_1)$ (otherwise x_2 would provide a shorter proof). By repeating this argument, we deduce that a shortest proof of $d(u, v) \geq \mathrm{rad}(G)$ uses a sequence $x_1, \ldots, x_p$ of nodes such that $d(v, x_i) \geq \mathrm{rad}(G) + d(u, x_1) + d(x_1, x_2) + \cdots + d(x_{i-1}, x_i)$ for $i = 1..p$ and $x_p \in L$. Consider the antipode $a = \mathrm{Antipode}_r(x_p)$. We then have $d(a, x_p) \geq d(v, x_p) \geq \mathrm{rad}(G) + d(u, x_1) + \cdots + d(x_{p-1}, x_p)$. By triangle inequality, we have $d(u, a) \geq d(a, x_p) - d(u, x_p)$ and $d(u, x_p) \leq d(u, x_1) + \cdots + d(x_{p-1}, x_p)$. We thus have $d(u, a) \geq \mathrm{rad}(G)$. In all cases, $L \cup \mathrm{Antipode}_r(L)$ must contain a node at distance $\mathrm{rad}(G)$ or more from u, it is thus a radius certificate. ∎

Note that a similar result does not hold for diameter certificate. A one-to-all distance based algorithm for diameter could query a set U of node such that for any pair u, v there is $x \in U$ satisfying $d(u, x) + d(x, v) \leq \mathrm{diam}(G)$ which implies $d(u, v) \leq \mathrm{diam}(G)$ by triangle inequality. Note that checking that a set U has this property requires quadratic time in general (under SETH) even if U has size $O(\log n)$ (see the reduction from SAT to diameter computation in [25]). Our diameter certificate definition requires that for each node u a single node x allows to bound all distances $d(u, v)$ for $v \in V$ using $d(u, v) \leq d(u, x) + e(x)$. The reason for this stronger requirement is to enable sub-quadratic time verification that a certificate is indeed a certificate when it has $o(n)$ size.

5 Radius computation and certification

We now propose a radius algorithm with complexity parametrized by the number of antipodes in the input graph. Similarly to previous algorithms [27, 5], it maintains lower bounds on eccentricities of all nodes and performs one-to-all distance queries from nodes with minimal lower bound as a first ingredient. Similarly to the two-sweeps and four-sweeps heuristics [20, 23, 8], it performs one-to-all distance queries from antipodes of previous query source as a second ingredient. Contrarily to these heuristics, it iterates until an exact solution is obtained (together with a radius certificate).

The idea of the algorithm is to maintain both a set K of nodes with distinct antipodes and a lower certificate L (initally empty). We iteratively select a node u with minimal lower-bound $e_L(u)$ and perform a one-to-all distance query from u. As long as this bound is not tight (i.e., $e_L(u) < e(u)$), we add $\mathrm{Antipode}_r(u)$ to L and u to K while eccentricity lower-bounds are improved accordingly. (The fact that the bound is not tight implies that no antipode of u could previously be in L.) As soon as the bound is tight (i.e., $e_L(u) = e(u)$), we then claim that u is a center (i.e., its eccentricity is minimal) and return $e(u)$ as the radius and L as radius certificate. Algorithm 1 formally describes the whole method.

Note the primal-dual flavor of this algorithm as the set K (which has same size as L) is a packing for $\{\mathrm{Antipode}_r^{-1}(u) : u \in V\}$ which is a restricted collection of $\{\overline{B}(u, \mathrm{rad}(G)) : u \in V\}$ for which the computed certificate L is a covering.

Input: A graph G and a ranking r of its node set V.
Output: The radius $\mathrm{rad}(G)$ of G, a center c and a radius certificate L.
$L := \emptyset$ `/* Lower certificate (tentative covering with` $\{\overline{B}(u, \mathrm{rad}(G)) : u \in V\}$`) */`
Maintain $e_L(v) = \max_{x \in L} d(v, x)$ (initially 0) for all $v \in V$.
$K := \emptyset$ `/* Packing for` $\{\mathrm{Antipode}_r^{-1}(u) : u \in V\}$`. */`
Do
  Select $u \in V$ such that $e_L(u)$ is minimal.
  $D_u := \mathrm{DistFrom}(G, u)$ `/* Distances from` u`. */`
  $e(u) := \max_{v \in V} D_u(v)$ `/* Eccentricity of` u`. */`
  If $e(u) = e_L(u)$ **then**
    return *$e(u)$, u, and L*
  else
    $a := \mathrm{argmax}_{v \in V}(D_u(v), r(v))$ `/* Antipode of` u `for` r`. */`
    $D_a := \mathrm{DistFrom}(G, a)$ `/* Distances from` a`. */`
    $K := K \cup \{u\}$
    $L := L \cup \{a\}$
    For $v \in V$ **do** $e_L(v) := \max(e_L(v), D_a(v))$
while $\min_{u \in V} e_L(u) < \min_{u \in K} e(u)$.
Return *$e(c)$, c and L where $c = \mathrm{argmin}_{u \in K} e(u)$.*

Algorithm 1: Computing the radius, a center and a radius certificate.

Theorem 3 *Given a graph G and a ranking r on its node set V, Algorithm 1 computes its radius $\mathrm{rad}(G)$, a center c and a radius certificate $L \subseteq \mathrm{Antipode}_r(V)$ with $2|L| + 1 = O(|\mathrm{Antipode}_r(V)|)$ one-to-all distance queries.*

Proof. We first prove the termination of Algorithm 1. Consider an iteration where we add the antipode a of the selected node u to L. We cannot have $a \in L$ as we would then have $e_L(u) = e(u)$ which is the termination case. In other words, nodes added to K have distinct antipodes and K is a packing for $\{\mathrm{Antipode}_r^{-1}(u) : u \in V\}$. As long as the do-while loop runs, each iteration adds a new node to L. If ever we reach the point where $L = \mathrm{Antipode}_r(V)$, then the lower-bound of each node $u \in V$ is tight: $e_L(u) = e(u)$. The next iteration must then terminate. The complexity is straightforward: at most $2|L| + 1$ one-to-all distance queries are performed and $|L| \leq |\mathrm{Antipode}_r(V)|$ as $L \subseteq \mathrm{Antipode}_r(V)$.

We now prove the correctness of Algorithm 1. Consider an iteration of the do-while loop. By the choice of u, we then have $e_L(v) \geq e_L(u)$ for all $v \in V$. If the termination case $e_L(u) = e(u)$ occurs, we have $e(v) \geq e_L(v) \geq e_L(u) = e(u)$ for all $v \in V$. This ensures that u has minimum eccentricity (it is a center). We thus have $\mathrm{rad}(G) = e(u)$ and L is a radius certificate as $e_L(v) \geq \mathrm{rad}(G)$ for all $v \in V$. Finally, if ever the condition for continuing the do-while loop is false, we have $\min_{u \in V} e_L(u) \geq \min_{u \in K} e(u)$. For $c = \mathrm{argmin}_{u \in K} e(u)$, we thus have $\mathrm{rad}(G) = \min_{u \in V} e(u) \geq \min_{u \in V} e_L(u) \geq e(c) \geq \mathrm{rad}(G)$. That is, L is a radius certificate and c is a center. ∎

In practice, we observe very fast convergence compared to $|\mathrm{Antipode}_r(V)|$ (see Section 9). We can give the following argument for that. The node u selected at each iteration satisfies $e_L(u) = \min_{v \in V} e_L(v) \leq \min_{v \in V} e(v) \leq \mathrm{rad}(G)$. We thus have $\max_{x \in L} d(u, x) \leq \mathrm{rad}(G)$, that is $u \in \cap_{x \in L} B[x, \mathrm{rad}(G)]$. It appears that the eccentricity of antipodes is generally large compared to radius in practical graphs, and the set $\cap_{x \in L} B[x, \mathrm{rad}(G)]$ tends to quickly shrink toward the set of centers as we add antipodes to L.

6 Minimum eccentricity selection

The core of the above radius algorithm is a general technique depending on a user-defined function f that we call *minimum eccentricity selection* (minES) for f. It is a procedure that returns a node with minimum eccentricity with respect to f. Its amortized complexity is low in graphs with few antipodes. More precisely, for a given graph G and a function f that maps a node v and an estimation ℓ of $e(v)$ to a value, it provides a function argminES returning a node u such that $f(u, e(u))$ is minimum as long as f is non-decreasing, i.e., $f(v, \ell) \leq f(v, \ell')$ for $\ell \leq \ell'$ for all v. A similar function minES returns the value of $f(u, e(u))$ for such node u. The challenge here is to avoid the computation of all eccentricities.

```
L := ∅   /* Lower certificate.                                  */
Maintain e_L(v) = max_{x∈L} d(v,x) (initially 0) for all v ∈ V.
Function argminES(G, r, L, e_L, f)
    Repeat
        u := argmin_{v∈V} f(v, e_L(v))
        D_u := DistFrom(G, u)     /* Distances from u.          */
        e(u) := max_{v∈V} D_u(v)  /* Eccentricity of u.         */
        If e_L(u) = e(u) then
            return u
        else
            a := argmax_{v∈V}(D_u(v), r(v))  /* Antipode of u for r. */
            D_a := DistFrom(G, a)   /* Distances from a.        */
            L := L ∪ {a}
            For v ∈ V do e_L(v) := max(e_L(v), D_a(v))
Function minES(G, r, L, e_L, f)
    u := argminES(G, r, L, e_L, f)
    Return f(u, e_L(u))
```

Algorithm 2: Minimum eccentricity selection with respect to function f.

We implement such a selection by maintaining lower bounds of all eccentricities as in Algorithm 1 and by using these lower bounds as estimates for true eccentricities. When the selection procedure is called, a node u which is minimum according to lower bounds is considered. Such a node is found by evaluating $f(v, e_L(v))$ for all $v \in V$ where $e_L(v)$ denotes the lower bound stored for a node v. A one-to-all distance query from u is then performed. If its eccentricity happens to be equal to its lower-bound $e_L(u)$ we claim that $f(u, e(u))$ is minimum and return that node. Otherwise, the antipode of u is used to improve lower bounds before trying again. Algorithm 2 formally describes this.

Proposition 1 *Given a graph G and a ranking r of its node set V, we consider a function f such that $f(v, \ell)$ can be evaluated for any $v \in V$ and $\ell \leq e(v)$. If $f(v, .)$ is non-decreasing for all $v \in V$, i.e., $f(v, \ell) \leq f(v, \ell')$ for $\ell \leq \ell'$, function* argminES *of Algorithm 2 returns a node u such that $f(u, e(u))$ is minimal and updates the lower certificate L such that $e_L(u) = e(u)$ and $f(v, e_L(v)) \geq f(u, e(u))$ for all $v \in V$. Moreover it can perform k computations of* $\operatorname{argmin} f(u, e(u))$ *using $k + 2|L'|$ one-to-all distance queries and $(k + 2|L'|)n$ calls to f where $L' \subseteq \mathrm{Antipode}_r(V)$ denotes the set of nodes added to L.*

Note that $(x + 2|L'|)n$ calls to f represent less than $O(x + 2|L'|)$ one-to-all distance queries with respect to time cost when f can be evaluated in constant time.

Proof. The correctness of the selection comes from the fact that $f(v, .)$ is non-decreasing: if $e_L(u) = e(u)$, we then have $f(u, e(u)) = f(u, e_L(u)) \leq \min_{v \in V} f(v, e_L(v)) \leq \min_{v \in V} f(v, e(v))$. The case $e_L(u) < e(u)$ can only occur if the antipode of u was not in L and happens at most $|\mathrm{Antipode}_r(V)|$ times in total. In particular, each call to argminES terminates. If an algorithm makes x calls to the argminES, the number of successful iterations where $e_L(u) = e(u)$ is precisely x while the number of unsuccessful iterations is at most the number of nodes added to L. For each such iteration we perform 2 one-to-all distance queries instead of 1. The total number of queries is thus $k + 2|L'|$. In all cases, we perform n calls to f per iteration. ∎

As an example of usage, Algorithm 1 for radius is equivalent to the following algorithm using our minimum eccentricity selection for the basic function $v, \ell \mapsto \ell$.

> $L := \emptyset$; $e_L(v) := 0$ for all $v \in V$.
> **Function** $\mathrm{ecc}(v, \ell)$*:* **return** ℓ
> $c := \mathrm{argminES}(G, r, L, e_L, \mathrm{ecc})$
> **return** $e_L(c), c, L$

As another example, the function f can be used to select a node with minimum eccentricity in a set W of nodes when $f(v, \ell)$ returns ℓ for $v \in W$ and ∞ otherwise. One can easily check that $f(v, .)$ is non-decreasing for all $v \in V$. We use our minimum eccentricity selection as an optimization for diameter computation and as a core tool for computing all eccentricities in the next sections.

7 Diameter computation and certification

We now analyze a simple diameter algorithm. The main ingredient of the algorithm consists in maintaining upper bounds of all eccentricities and performing one-to-all distance queries from nodes with maximum upper bound. It thus follows the main line of previous practical algorithms [27, 5]. However, we present the algorithm with a more general primal-dual approach which was not noticed before. Moreover, we introduce a new technique called *delegate certificate*: after selecting a node u with maximal upper bound, it consists in performing a one-to-all distance query from any node x such that $(d(u, x) + e(x) = e(u)$. We call such a node x a *tight upper certificate* for u as we have $e^{\{x\}} = e(u)$. A possible choice for x is u itself in which case the algorithms becomes a very basic version of [27]. However, we observe that choosing a node x with minimal eccentricity offers much better performances in practice (see Section 9). Our complexity analysis is independent of the choice of x, we thus present the algorithm in the most general manner.

The algorithm grows both a packing K and an upper certificate U until the upper bound $e^U(u)$ on the eccentricity of any node u is at most the maximum eccentricity of nodes in K. As long as this condition is not satisfied, a node u with maximal upper bound is selected and added to K. We then choose a tight upper certificate x for u and add it to U. Note that we now have $e^U(u) = e(u) \leq \max_{v \in K} e(v)$ and u cannot be selected again. This ensures that the termination condition is reached at some point when U is a certificate that all nodes have eccentricity at most that of a maximum eccentricity node in K which must thus be equal to diameter. See Algorithm 3 for a formal description.

We claim that the set K is a packing for the collection $\mathcal{D}_{1/3} = \{B(u, \frac{1}{3}(\mathrm{diam}(G) - e(u))) : u \in V\}$ of open balls. As it has same size as the certificate U returned by the algorithm in the end, this allows to state the following theorem.

Theorem 4 *Given a graph G, Algorithm 3 computes the diameter of G, a diametral node b and a diameter certificate U of size $\pi_{1/3}$ at most with $O(\pi_{1/3})$ one-to-all distance queries where π_α is the maximum packing size for the collection of open balls $\mathcal{D}_\alpha = \{B(u, \alpha(\mathrm{diam}(G) - e(u))) : u \in V\}$ for $\alpha > 0$. It approximates minimum diameter certificate within a factor $\frac{\pi_{1/3}}{\pi_{[1]}}$ where $\pi_{[1]}$ is the maximum packing size for the collection $\mathcal{D}_{[1]} = \{B[u, \mathrm{diam}(G) - e(u)] : u \in V\}$.*

Input: A graph G and a ranking r of its node set V.
Output: The diameter $\mathrm{diam}(G)$ of G, a diametral node b and a diameter certificate U.
$U := \emptyset$ `/* Upper certificate (tentative covering with` $\{B[u, \mathrm{diam}(G) - e(u)] : u \in V\}$`). */`
Maintain $e^U(u) = \min_{x \in U} d(u, x) + e(x)$ (initially ∞) for all $v \in V$.
$K := \emptyset$ `/* Packing for` $\{B(u, \frac{1}{3}(\mathrm{diam}(G) - e(u))) : u \in V\}$`. */`
Do
  Select u such that $e^U(u)$ is maximal.
  $D_u := \mathrm{DistFrom}(G, u)$ `/* Distances from` u`. */`
  $e(u) := \max_{v \in V} D_u(v)$ `/* Eccentricity of` u`. */`
  $K := K \cup \{u\}$
1 Select x such that $d(u, x) + e(x) = e(u)$. `/* Delegate certificate for` u`. */`
2 $D_x := \mathrm{DistFrom}(G, x)$ `/* Distances from` x`. */`
  $e(x) := \max_{v \in V} D_x(v)$ `/* Eccentricity of` x`. */`
  $U := U \cup \{x\}$
  For $v \in V$ **do** $e^U(v) := \min(e^U(v), D_x(v) + e(x))$
while $\max_{u \in K} e(u) < \max_{u \in V} e^U(u)$
Return $e(b)$, *b and U where $b \in K$ satisfies $e(b) = \max_{u \in K} e(u)$.*

Algorithm 3: Computing diameter and a diameter certificate. The basic version of the algorithm consists in selecting $x := u$ in Line 1. (The redundant one-to-all distance query in Line 2 can then be omitted.)

Proof. We already argued the termination and the correctness of the algorithm above. We thus show the packing property of K. Suppose for the sake of contradiction that K is not a packing for $\mathcal{D}_{1/3}$. Consider the first iteration where a node v is added to K while some open ball $B(y, \frac{1}{3}(\mathrm{diam}(G) - e(y)))$ in $\mathcal{D}_{1/3}$ contains both v and some node $u \in K$ added previously. Let x be the tight upper certificate for u that was added to U. By triangle inequality, we have $d(x, v) \leq d(x, u) + d(u, y) + d(y, v)$. The choice of x implies $d(x, u) = e(u) - e(x) \leq d(u, y) + e(y) - e(x)$. Combining the two inequalities, we obtain $d(x, v) \leq 2d(u, y) + d(y, v) + e(y) - e(x)$. As u and v are in $B(y, \frac{1}{3}(\mathrm{diam}(G) - e(y)))$, we have $2d(u, y) + d(y, v) < \mathrm{diam}(G) - e(y)$ and finally get $d(x, v) < \mathrm{diam}(G) - e(x)$ which implies $e^U(v) < \mathrm{diam}(G)$. However, it is required that v has maximal upper bound $e^U(v) = \max_{w \in V} e^U(w)$ when it is selected for being added to K, in contradiction with $\max_{w \in V} e^U(w) \geq \max_{w \in V} e(w) = \mathrm{diam}(G)$. We conclude that K must be a packing for $\mathcal{D}_{1/3}$. Both sizes of K and U are bounded by $\pi_{1/3}$. As any diameter certificate is a covering for $\mathcal{D}_{[1]}$ and has size $\pi_{[1]}$ at least, this guarantees that the size of U is within a factor $\frac{\pi_{1/3}}{\pi_{[1]}}$ at most from optimum. ∎

This analysis can be complemented when we start Algorithm 3 with $K := \{u\}$ and $U := \{c\}$ initially where c is a center of the graph computed with Algorithm 1. We reference this combination as Algorithm 1+3 in the sequel. A similar proof then allows to show that K is a packing for

$\mathcal{D}^c_{1/3}(G) = \{B(u, \beta_u(\mathrm{diam}(G) - e(u))) : u \in V\}$ where $\beta_u = 1/3$ for $u \neq c$ and $\beta_c = 1$. We obtain the following corollary from Theorems 3 and 4.

Corollary 1 *Given a graph G and a ranking r of its node set V, Algorithm 1+3 computes the diameter of G, a diametral node b and a diameter certificate U of size $\pi^c_{1/3}$ at most with $|U| + 2\,|\mathrm{Antipode}_r(V)| + 1$ one-to-all distance queries at most where c is a center of G returned by Algorithm 1 and $\pi^c_{1/3} = \pi(\mathcal{D}^c_{1/3})$ is the maximum packing size for the collection $\mathcal{D}^c_{1/3} = \{B(u, \beta_u(\mathrm{diam}(G) - e(u))) : u \in V\}$ of open balls with radii factors $\beta_u = \frac{1}{3}$ for $u \neq c$ and $\beta_c = 1$ (for $u = c$).*

Proof. In addition to the proof of Theorem 4, we just have to consider the case when a node v would be added to K while having $v \in B(c, \mathrm{diam}(G) - e(c))$. As $c \in U$, we then have $e^U(v) \leq d(c, v) + e(c) < \mathrm{diam}(G)$. This would raise a contradiction as the choice of v relies on $e^U(v) = \max_{w \in V} e^U(w) \geq \max_{w \in V} e(w) \geq \mathrm{diam}(G)$. ∎

This explains efficiency of practical algorithms as we observe that coverings of small size often exist for $\mathcal{D}^c_{1/3}(G)$ in practical graphs (see Section 9). As mentioned before, a further optimization consists in selecting a tight upper certificate x for u with minimal eccentricity. Using a function f such that $f(v, \ell)$ returns ℓ when $D_u(v) + \ell \leq e(u)$ and returns ∞ otherwise, it can be obtained through our minimum eccentricity selection procedure by replacing Line 1 with $x := \mathrm{argminES}(G, r, L, e_L, f)$. The algorithm is referenced as Algorithm 1+3' in the sequel. This optimization through the delegate certificate technique provides performances similar to previous practical algorithms (see Section 9) while providing the above complexity guarantees (Corollary 1 also applies to this variant).

8 All eccentricities

We now present a novel algorithm for all eccentricities. It relies on minimum eccentricity selection and properties of tight upper certificates.

8.1 Optimal tight upper certificate

We first characterize the minimum tight upper certificate of a graph G which is tightly related to the notion of tight upper certificate.

Proposition 2 *Given a graph G, being a tight upper certificate defines a binary relation $\preceq$ which is a partial order ($u \preceq x$ stands for $e(u) = d(u, x) + e(x)$). Moreover, the set $U^{\preceq}$ of all maximum elements of this partial order is the unique tight upper certificate of G with minimum size.*

Proof. We first prove that the relation $\preceq$ is a partial order. It is obviously reflexive as the distance from a node to itself is zero, implying $e(u) = d(u, u) + e(u)$. It is antisymmetric: if x and y are both tight upper certificates one for the other, we then have $e(x) = d(x, y) + e(y)$ and $e(y) = d(y, x) + e(x)$, and thus $d(x, y) = 0$. We finally show transitivity. Suppose that y is a tight upper certificate for x and that z is an tight upper certificate for y, that is $e(x) = d(x, y) + e(y)$ and $e(y) = d(y, z) + e(z)$. We thus have $d(x, y) + d(y, z) + e(z) = e(x)$. As triangle inequality implies $e(x) \leq d(x, z) + e(z)$, we obtain $d(x, y) + d(y, z) \leq d(x, z)$, and thus $d(x, y) + d(y, z) = d(x, z)$ by triangle inequality again. We finally get $e(x) = d(x, z) + e(z)$ and z is a tight upper certificate for x.

We now show that the set $U^{\preceq}$ of maximal elements for $\preceq$ is the unique optimal tight upper certificate of G. For any non-maximal element u, we can build a chain $u \preceq x_1 \preceq x_2 \preceq \cdots$ where u has a tight upper certificate x_1, if x_1 is not in $U^{\preceq}$, it has a tight upper certificate x_2, and so on. As the partial order is finite, the chain must be finite and x_k must be in $U^{\preceq}$ for some k. The transitivity of $\preceq$ implies that x_k is a tight upper certificate for u implying $e^{U^{\preceq}}(u) \leq d(u, x_k) + e(x_k) = e(u)$.

This shows that $U^{\preceq}$ is a tight upper certificate. As each element x of $U^{\preceq}$ is the only tight upper certificate for itself (as a maximal element), $U^{\preceq}$ is included in any tight upper certificate of G. In particular, any minimum tight upper certificate must indeed equal $U^{\preceq}$. ∎

Note that $U^{\preceq}$ includes in particular all centers of the graph: a center c cannot have a tight upper certificate $x \neq c$ (otherwise we have $e(x) = e(c) - d(c,x) < e(c)$ in contradiction with the minimality of $e(c)$).

8.2 All eccentricity computation and certification

We now propose to compute all eccentricities of a graph as follows (see Algorithm 4 for a formal description). We maintain both a lower certificate L and an upper certificate U. As long as some node has untight upper bound, we select a node u with untight upper bound and minimal eccentricity using our minimum eccentricity selection procedure which then additionally ensures $e_L(u) = e(u)$. (We use for that purpose a function returning ∞ when the eccentricity value equals the upper bound.) We claim that u is in $U^{\preceq}$ (see Lemma 1 bellow). We thus add u to the upper certificate U and update upper bounds accordingly. When our minimum eccentricity selection procedure detects that all nodes have tight upper bounds, lower bounds must be tight also. The algorithm then terminates with the following guarantees.

Input: A graph G and a ranking r of V.
Output: All eccentricities, a tight lower certificate L of G and a tight upper certificate U of G.
$L := \emptyset$ `/* Lower certificate (tentative hitting set for` $\{\overline{B}(u, e(u)) : u \in V\}$`) */`
Maintain $e_L(v) = \max_{x \in L} d(v, x)$ (initially 0) for all $v \in V$.
$U := \emptyset$ `/* Upper certificate (maximal nodes for` $\preceq$`) */`
Maintain $e^U(u) = \min_{x \in U} d(u, x) + e(x)$ (initially ∞) for all $v \in V$.
Function ecc-untight(v, ℓ)
  If $\ell < e^U(v)$ **then return** ℓ **else return** ∞
While minES$(L, e_L, \text{ecc-untight}) < \infty$ **do**
  $u :=$ argminES$(G, r, L, e_L, \text{ecc-untight})$
  $D_u :=$ DistFrom(G, u) `/* Distances from` u`. */`
  $e(u) := \max_{v \in V} D_u(v)$ `/* Eccentricity of` u`. */`
  $U := U \cup \{u\}$
  For $v \in V$ **do** $e^U(v) := \min(e^U(v), D_u(v) + e(u))$
Return e^U, L, U

Algorithm 4: Computing all eccentricities and tight lower/upper certificates.

Theorem 5 *Given a graph G and a ranking r of its node set V, Algorithm 4 computes all eccentricities, a tight lower certificate $L \subseteq \text{Antipode}_r(V)$ and the optimal tight upper certificate $U^{\preceq}$ with $|U^{\preceq}| + 2|L|$ one-to-all distance queries.*

In practical graphs, we observe that the size of $U^{\preceq}$ is much larger than that of the computed lower certificate L and the algorithm roughly costs $|U^{\preceq}|$ BFS traversals (see Section 9).

The correctness of Algorithm 4 mainly rely on the following lemma.

Lemma 1 *Consider an upper certificate U and the set S_U of nodes that do not have a tight upper certificate in U. Any node $v \in S_U$ with minimum eccentricity (having $e(v) = \min_{u \in S_U} e(u)$) is its unique tight upper certificate (i.e., $v \in U^{\preceq}$).*

Proof. For the sake of contradiction, suppose that v has a tight upper certificate $x \neq v$. As $e(v) = d(v,x) + e(x)$, we have $e(x) < e(v)$ and x cannot be in S_U by the minimality of $e(v)$. It thus has a tight upper certificate $y \in U$. But the transitivity of being a tight upper certificate (Proposition 2) implies that y is also a tight upper certificate for v which is in contradiction with

type	name	n	m/n	d	w	$\frac{\text{diam}}{\text{rad}}$	D	$\pi^c_{0.8}$	$\pi^c_{1/3}$	n_c/n	R	A_{ID}	F
comm	Gnutella	14149	3.60	•	○	1.58	19	47	1912	0.27	4	10	23
comm	skitter	1694616	13.09	○	○	1.94	3	5	7	0.99	3	6	6
game	FrozenSea	753343	7.70	○	•	1.80	15	35	191	0.91	7	384	388
geom	buddha	543652	6.00	○	•	1.87	27	63	385	0.95	14	897	897
road	CAL-d	1890815	2.45	○	•	1.89	3	17	90	0.99	3	11	11
road	CAL-t	1890815	2.45	○	•	1.83	7	17	105	0.98	5	13	13
road	CAL-u	1890815	2.45	○	○	1.99	2	4	6	0.99	3	7	11
road	FLA-t	1070376	2.51	○	•	1.99	2	4	4	0.99	2	2	2
road	europe-t	18010173	2.34	•	•	1.99	2	4	5	1	2	-	-
soc	Epinions	32223	13.76	•	○	2	2	4	7	0.99	2	7	20
soc	Hollywood	1069126	106.33	○	○	1.71	29	603	4183	0.99	3	34	335
soc	Slashdot	71307	12.80	•	○	1.86	4	40	65	0.99	4	12	135
soc	Twitter	68413	24.63	•	○	2.50	2	6	8	0.99	4	31	4753
soc	dblp	226413	6.33	○	○	2	1	11	20	1	5	6	43
synth	bowtie500	505002	2.00	○	○	1.99	3	2001	4001	0.99	5	5	1507
synth	grid1500-wd	296680	1.66	•	•	2.38	3	12	69	0.04	4	5	6
synth	grid500-10	250976	3.59	○	○	2	1	5	5	1	3	4	4
synth	pwlaw2.5	1000000	3.85	○	○	1.90	4	24	24	0.99	2	19	47
synth	udg10	999888	9.99	○	○	1.99	5	17	86	0.99	4	5	10
vlsi	alue7065	34046	3.22	○	○	2	1	5	5	1	3	4	4
web	BerkStan	334857	13.51	•	○	2.73	2	7	28	2e-03	3	16	17
web	Indochina	3806327	25.96	•	○	6.91	2	4	6	0.99	3	-	-
web	NotreDame	53968	5.65	•	○	2.11	2	5	22	0.01	2	2	2

Table 1: Diameter and radius certificate sizes (D, R) for various graphs, and related parameters.

$v \in S_U$. ∎

Proof.[of Theorem 5] We first prove $U \subseteq U^{\preceq}$. As in Lemma 1, let S_U denote the set of nodes that do not have a tight upper certificate in U ($S_U = \{v \in V \mid e(v) < e^U(v)\}$). Now consider the node u selected at some iteration of the while loop. We prove $u \in U^{\preceq} \setminus U$. The correctness of our minimum eccentricity selection (Proposition 1) implies that u has minimum eccentricity in S_U and is thus in $U^{\preceq}$ by Lemma 1. (Note that ecc-untight$(v, .)$ is non-decreasing for all $v \in V$ as ecc-untight$(v, \ell) = \ell$ for $\ell < e(v)$ and ecc-untight$(v, \ell) = \infty$ for $\ell \geq e(v)$.) Additionally, $u \in S_U$ implies that u has untight upper bound and is not in U until we add it at that iteration.

The termination of the algorithm is guaranteed by the fact that U grows at each iteration. The algorithm ends when minimum eccentricity selection returns a node u such that ecc-untight$(u, e(u)) = \infty$. Proposition 1 then ensures ecc-untight$(v, e_L(v)) = \infty$ for all v. That is $e_L(v) = e^U(v)$ for all v and both bounds must equal $e(v)$. This implies that L (resp. U) is then a tight lower (resp. upper) certificate of G. Moreover, $U \subseteq U^{\preceq}$ then implies $U = U^{\preceq}$ by Proposition 2. ∎

9 Experiments

We test social networks (Epinions, Hollywood, Slashdot, Twitter, dblp), computer networks (Gnutella, Skitter), web graphs (BerkStan, IndoChina, NotreDame), road networks (CAL-t, CAL-d, CAL-u, FLA-t, europe-t), a 3D triangular mesh (buddha), and grid like graphs from VLSI applications (alue7065) and from computer games (FrozenSea). The data is available from `snap.stanford.edu`, `webgraph.di.unimi.it`, `www.dis.uniroma1.it/challenge9`, `graphics.stanford.edu`, `steinlib.zib.de` and `movingai.com`. We also test synthetic inputs: bowtie500 is the graph $BT_{500,500}$ represented in Figure 1, grid500-10 is a 501×501 square grid with random deletion of 10% of the edges, grid1500-wd is a weighted directed graph obtained from a 1501×1501 square grid where each edge is oriented randomly (with probability 1/2 for each direction) and assigned a random weight uniformly in $\{0, 1, \ldots, 9\}$, pwlaw2.5 is a random powerlaw graph where the number of nodes of degree k is proportional to $k^{-2.5}$, udg10 is a random unit disk graph where field size is parametrized to obtain average degree 10 roughly. Each graph is restricted to its largest (strongly) connected component.

Table 1 summarizes our main practical observations. For each instance G, we show its type, the number n of nodes in the largest (strongly) connected component, the average out-degree m/n, whether it is directed (d) and weighted (w), and the diameter to radius ratio $\frac{\mathrm{diam}(G)}{\mathrm{rad}(G)}$. We then show the size D of the diameter certificate computed by Algorithm 1+3', bounds on maximum packing sizes $\pi^c_{0.8}$ and $\pi^c_{1/3}$ (defined in Section 7), the proportion n_c/n of nodes in the (in-)ball of radius $\mathrm{diam}(G) - \mathrm{rad}(G)$ centered at a center c, the size R of the radius certificate computed by Algorithm 1, the number A_{ID} of antipodes for ID ranking and the number F of furthest nodes. The two latter numbers were obtained by performing a traversal per node of the graph (in quadratic time). A dash indicates a value that could not be obtained in less than few days of computation.

The first observation is that diameter and radius certificates are extremely small for all instances (less than 30 nodes for all of them). Several observations allow to explain this phenomenon. First, all graphs have high diameter to radius ratio (over 1.5 for all of them). Note that this ratio is at most 2 for an undirected graph (it is unbounded in general directed graphs). Undirected graphs with ratio 2 have a one node diameter certificate: a center. This concerns two practical graphs while several ones have ratio very close to 2. Coherently, the concentration of nodes around the center is also high with respect to the diameter minus radius difference. This is measured by the ratio n_c/n where c is a center computed by our radius algorithm ($e(c) = \mathrm{rad}(G)$) and n_c denotes the number of nodes in $B[c, \mathrm{diam}(G) - \mathrm{rad}(G)]$ (in directed graphs we count the number of nodes u such that $d(u, c) \leq \mathrm{diam}(G) - \mathrm{rad}(G)$). It counts the proportion of nodes u such that $e^{\{c\}}(u) \leq \mathrm{diam}(G)$ which appears to be very close to 1 for most of the graphs. Notable exceptions are Gnutella, BerkStan and NotreDame. This may be explained by the low (compared to others) diameter to radius ratio (1.58) of the first one and probably to the highly asymmetric nature of the two others. Seeing diameter certification as a covering problem with balls $B[x, \mathrm{diam}(G) - e(x)]$, there are thus few nodes that are not covered by a center c. Additionally, other nodes can be covered using few balls with reduced radii: the columns $\pi^c_{0.8}$ and $\pi^c_{1/3}$ indicate the size of coverings we could find using balls with radii reduced by a factor .8 and 1/3 respectively. These numbers upper bound maximum packing sizes $\pi^c_{0.8}$ and $\pi^c_{1/3}$ of the associated collections of balls with reduced radii. Our theoretical upper-bound of $\pi^c_{1/3}$ thus explains fast diameter computation for most of the graphs. A notable exception is $BT_{500,500}$ alias bowtie500 which was tailored for making former diameter algorithms slow (including Algorithm 1+3) and thus have large $\pi^c_{1/3}$ value. Other exceptions are Hollywood and Gnutella for which the diameter to radius ratios are not so high either (compared to other graphs). However the parameter is still much smaller than the number of nodes.

Concerning radius computation, we observe that most graphs have very few antipodes as indicated by the A_{ID} column although the number F of furthest nodes can be much larger (as in the Twitter graph). A notable exception is the buddha graph which is a triangulated 3D surface and thus has more or less a sphere like topology (the arms form handles) that may explain why all furthest nodes are antipodes. However the number of antipodes remains much smaller than the number of nodes. ForzenSea also has a relatively large number of furthest nodes that are almost all antipodes. This might come from the design of the graph as a map where players of a video game evolve and should find dead ends.

10 Graphs with low doubling dimension

A graph is γ-doubling if every ball is included in the union of at most γ balls with half radius.

10.1 Exact diameter computation

In the case of γ-doubling graphs, the following theorem shows that we can obtain a diameter certificate with linear size compared to the maximum packing size $\pi(\mathcal{D}_\alpha)$ for the collection $\mathcal{D}_\alpha$ (see Section 7). This complements Theorem 4 in the range $\frac{1}{3} \le \alpha < 1$.

Theorem 6 *Given a γ-doubling graph G and $\alpha < 1$, the diameter $\operatorname{diam}(G)$, a diametral node p and a diameter certificate U satisfying $|U| \le \pi_\alpha \gamma^{O(1)+\log\frac{1}{1-\alpha}} \log \operatorname{diam}(G)$ can be computed with $2\,|\mathrm{Antipode}(V)| + |U|$ one-to-all distance queries where π_α is the maximum size of a packing for the collection $\mathcal{D}_\alpha = \{B(u, \alpha(\operatorname{diam}(G) - e(u))) : u \in V\}$.*

Note that this implies that minimum diameter certificate can be approximated within a factor $\gamma^{O(1)+\log(\operatorname{diam}(G)-\operatorname{rad}(G))} \frac{\pi_1}{\pi_{[1]}}$ when G is γ-doubling as $\mathcal{D}_\alpha = \mathcal{D}_1$ for $\alpha > 1 - \frac{1}{r+1}$ where $r = \operatorname{diam}(G) - \operatorname{diam}(R)$ is the maximum radius of a ball in $\mathcal{D}_1$ (recall that $\pi_{[1]}$ lower bounds the size of a minimum diameter certificate).

The above theorem is a consequence of Algorithm 5 which follows a primal-dual approach by constructing both a packing K for $\mathcal{D}_\alpha$ together with a covering U with $\mathcal{D}_{[1]}$ such that $|U| \le |K|\,\gamma^{O(1)+\log\frac{\alpha}{1-\alpha}} \log \operatorname{diam}(G)$. Given a node u, let $S_u = \{v \in V \mid \exists B \in \mathcal{D}_\alpha \text{ s.t. } u, v \in B\}$ denote the set of nodes v that cannot be in packing for $\mathcal{D}_\alpha$ containing u. The idea is to iteratively add a node u to K with highest eccentricity according to e^U and then to add sufficiently many nodes to U so that any node v in S_u gets an eccentricity upper bound $e^U(v)$ equal to $\operatorname{diam}(G)$ or less. This will guarantee that no such node is added later to K and that K is a packing for $\mathcal{D}_\alpha$.

Input: A graph G and a parameter α with $0 < \alpha < 1$.
Output: The diameter $\operatorname{diam}(G)$ of G and a diameter certificate p, U.

```
K := ∅  /* Packing for {B(u, α(diam(G) − e(u))) : u ∈ V}.          */
U := ∅  /* Upper certificate.                                       */
Maintain e^U(u) = min_{x∈U} d(u,x) + e(x) (initially ∞) for all v ∈ V.
L := ∅  /* Lower certificate.                                       */
Maintain e_L(v) = max_{x∈L} d(v,x) (initially 0) for all v ∈ V.
While max_{p∈K∪P} e(p) < max_{u∈V} e^U(u) do
    Select u such that e^U(u) is maximal.
    D_u := DistFrom(u)
    e(u) := max_{w∈V} D_u(w)   /* Eccentricity of u.                 */
    K := K ∪ {u}
    U := U ∪ {u}
    For w ∈ V do e^U(w) := min(e^U(w), D_u(w) + e(u))
    Function ecc-slack-far(v, ℓ)
        If e^U(v) − ℓ > (1−α)/(2α) D_u(v) then return −D_u(v) else return ∞
    While minES(G, r, L, e_L, ecc-slack-far) < ∞ do
        v := argminES(G, r, L, e_L, ecc-slack-far)
        D_v := DistFrom(v)
        e(v) := max_{w∈V} D_v(w)   /* Eccentricity of v.             */
        U := U ∪ {v}
        For w ∈ V do e^U(w) := min(e^U(w), D_v(w) + e(v))
p := argmax_{p∈K∪P} e(p)
Return e(p) and p, U.
```

Algorithm 5: Computing diameter and a diameter certificate assuming doubling property.

Our selection rule for adding nodes to U is based on comparing their eccentricity to their distance to u. The rough idea is to add a node v as certificate in U when $e^U(v) > \operatorname{diam}(G)$ and $\operatorname{diam}(G) - e(v) = \Omega(d(u, v))$. Note that any node $w \in B[v, \operatorname{diam}(G) - e(v)]$ then satisfies $e^U(w) \le \operatorname{diam}(G)$ and the doubling property will allow us to bound the number of nodes added to U. As the eccentricity of v is not known precisely until we perform a one-to-all distance query from

v, we use our minimum eccentricity selection technique using a lower certificate L. As $\mathrm{diam}(G)$ is not known either, we select v such that $e^U(v) - e(v) = \Omega(d(u,v))$. This ensures that a ball of radius $\Omega(d(u,v))$ will then be covered, we thus prefer v such that $d(u,v)$ is additionally maximal.

Proof.[of Theorem 6] The main arguments of the proof are the following.

The function $\ell \mapsto \text{ecc-slack-far}(v,\ell)$ is non-decreasing as it returns $-d(u,v)$ for $\ell <= e^U(v) - \frac{1-\alpha}{2\alpha} d(u,v)$ and ∞ otherwise. Note that the minimum eccentricity selection for ecc-slack-far thus returns a node v such that $e^U(v) - e(v) \geq \frac{1-\alpha}{2\alpha} d(u,v)$ and $d(u,v)$ is maximal.

We first prove that the inner loop performs at most $\gamma^{O(1)+\log\frac{1}{1-\alpha}} \log \mathrm{diam}(G)$ iterations. This bounds the number of nodes added to U when one node is added to K and will thus ensures $|U| \leq |K| \gamma^{O(1)+\log\frac{1}{1-\alpha}} \log \mathrm{diam}(G)$. Consider an iteration of the main loop where u is added to K. Just after adding v to U in the inner loop, consider w s.t. $d(v,w) \leq \frac{\rho}{2+\rho} d(u,v)$ where $\rho = \frac{1-\alpha}{2\alpha}$. We then have $e^U(w) \leq d(v,w) + e(v)$ as $v \in U$ and $e(w) \geq e(v) - d(v,w)$ by triangle inequality. This gives $e^U(w) - e(w) \leq \frac{2\rho}{2+\rho} d(u,v)$. As $d(u,w) \geq d(u,v) - d(v,w) \geq \frac{2}{2+\rho} d(u,v)$, we get $e^U(w) - e(w) \leq \rho d(u,w)$ and nodes in $B[v, \frac{\rho}{2+\rho} d(u,v)]$ do not satisfy the condition of the inner loop. The doubling property implies that the number of iterations where we select v such that $d(u,v) > \frac{e(u)}{2}$ is at most $\gamma^{O(1)+\log\frac{1}{1-\alpha}}$. Then we may select v such that $d(u,v) > \frac{e(u)}{4}$ during the same number of iterations at most, and so on until we eventually select u itself (the only node v such $d(u,v) = 0$). The overall number of iterations of the inner loop is thus bounded by $\gamma^{O(1)+\log\frac{1}{1-\alpha}} \log e(u)$.

For the sake of contradiction, suppose that K is not a packing and consider $u, u' \in K$ and $x \in V$ such that both u and u' are in $B(x, \alpha(\mathrm{diam}(G) - e(x)))$. Assume without loss of generality that u was added to K before u'. After the inner loop for u, we have $e^U(x) - e(x) \leq \frac{1-\alpha}{2\alpha} d(u,x)$. There thus exists $y \in U$ such that $d(x,y) + e(y) \leq e(x) + \frac{1-\alpha}{2\alpha} d(u,x)$. We thus have $e^U(u') \leq d(u',y) + e(y) \leq d(u',x) + d(x,y) + e(y) \leq d(u',x) + e(x) + \frac{1-\alpha}{2\alpha} d(u,x)$. As $u, u' \in B(x, \alpha(\mathrm{diam}(G) - e(x)))$, we get $e^U(u') < \frac{1+\alpha}{2} \mathrm{diam}(G) + \frac{1-\alpha}{2} e(x)$. As $e(x) \leq \mathrm{diam}(G)$, we obtain $e^U(u') < \mathrm{diam}(G)$. This is a contradiction since the choice of u' implies $e^U(u') = \max_{v \in V} e^U(v) \geq \mathrm{diam}(G)$. ∎

10.2 Approximating radius and diameter

Interestingly, the following lemma links the gap between the bound provided by a lower/upper certificate for a node u and the distance from the certificate to a tight lower/upper certificate for u. Recall that a tight upper certificate for u is a node x such that $e(u) = d(u,x) + e(x)$. We similarly define a tight lower certificate for u as a node x such that $e(u) = d(u,x)$ (equivalently, x is a furthest node from u).

Lemma 2 *Given a lower certificate L (resp. an upper certificate U) and a node u, we have $e(u) - e_L(u) \leq d(x,L)$ (resp. $e^U(u) - e(u) \leq 2d(x,U)$) for any tight lower (resp. upper) certificate x for u.*

Proof. Consider a tight lower certificate x for a node u ($e(u) = d(u,x)$). Let $y \in L$ be the closest node to x in L ($d(x,y) = d(x,L)$). By triangle inequality, we have $e(u) = d(u,x) \leq d(u,y) + d(y,x) \leq e_L(u) + d(x,L)$.

Similarly, consider a tight upper certificate x for a node u ($e(u) = d(u,x) + e(x)$). Let $y \in U$ be the closest node to x in U ($d(x,y) = d(x,U)$). By triangle inequality, we have $e(u) = d(u,x) + e(x) \geq d(u,y) - d(x,y) + e(y) - d(x,y) \geq e^U(u) - 2d(x,U)$. ∎

Now consider the choice of a node u with minimal eccentricity lower bound in Algorithm 1. This choice implies $e_L(u) \leq \mathrm{rad}(G)$ and Lemma 2 then implies $e(u) \leq \mathrm{rad}(G) + d(a,L)$ where a is the antipode of u which is added to L (if u is not a center). As long as the selected node has eccentricity greater than $(1+\varepsilon)\,\mathrm{rad}(G)$, the nodes in L are $\varepsilon\,\mathrm{rad}(G)$ far apart and form a packing for the collection of balls of radius $\varepsilon\,\mathrm{rad}(G)/2$. Similarly, the choice of a node u with maximal eccentricity upper bound in Algorithm 3 implies $e^U(u) \geq \mathrm{diam}(G)$ and Lemma 2 then implies $e(u) \geq \mathrm{diam}(G) - 2d(x, U)$ where x is the tight upper certificate chosen for u that is added to U. As long as the selected node has eccentricity less than $(1-\varepsilon)\,\mathrm{diam}(G)$, the nodes in U form a packing for the collection of balls of radius $\frac{\varepsilon}{2}\,\mathrm{diam}(G)/2$. As the doubling property implies that such packings have size bounded by $\gamma^{\lceil \log \frac{2}{\varepsilon} \rceil + 1}$, we obtain the following approximation results for radius and diameter.

Proposition 3 *Given a γ-doubling graph G and $\varepsilon > 0$, Algorithm 1 (resp. Algorithm 3) provides a node u with eccentricity $(1+\varepsilon)\,\mathrm{rad}(G)$ at most (resp. $(1-\varepsilon)\,\mathrm{diam}(G)$ at least) with $O(\gamma^{\lceil \log \frac{2}{\varepsilon} \rceil + 1})$ traversals.*

Proof. As discussed above, the set L is a packing for balls of radius $\varepsilon\,\mathrm{rad}(G)$ until a node whose eccentricity approximates the radius is found. We use the fact that packing size is bounded by covering size. By the γ-doubling property, the whole graph can be covered by γ^i balls of radius $\frac{2\,\mathrm{rad}(G)}{2^i}$ as any ball of radius $2\,\mathrm{rad}(G)$ contains all nodes. For $i \geq \log \frac{2}{\varepsilon} + 1$ these balls have radius $\varepsilon\,\mathrm{rad}(G)/2$ at most. Using that packing size is bounded by covering size, we can bound the number of iterations of Algorithm 1 where chosen nodes u have eccentricity greater than $(1+\varepsilon)\,\mathrm{rad}(G)$. If we stop the algorithm after $\gamma^{\lceil \log \frac{2}{\varepsilon} \rceil + 1}$ iterations, the node $u \in K$ with smallest eccentricity is guaranteed to have eccentricity $(1+\varepsilon)\,\mathrm{rad}(G)$ at most. The argument for diameter approximation is similar. ∎

11 Negatively curved graphs

In this section we analyze the behavior of our algorithms in the class of negatively curved graphs, alias, δ-hyperbolic graphs.

Let (X, d) be a metric space and $w \in X$. The *Gromov product* of $y, z \in X$ with respect to w is defined to be

$$(y|z)_w = \frac{1}{2}(d(y, w) + d(z, w) - d(y, z)).$$

Let $\delta \geq 0$. A metric space (X, d) is said to be δ-*hyperbolic* [18] if

$$(x|y)_w \geq \min\{(x|z)_w, (y|z)_w\} - \delta$$

for all $w, x, y, z \in X$. Equivalently, (X, d) is δ-hyperbolic if for any four points u, v, x, y of X, the two larger of the three distance sums $d(u, v) + d(x, y)$, $d(u, x) + d(v, y)$, $d(u, y) + d(v, x)$ differ by at most $2\delta \geq 0$. A graph $G = (V, E)$ is δ-hyperbolic if the associated shortest path metric space (V, d) is δ-hyperbolic.

From the definition of a δ-hyperbolic graph, we immediately get the following simple but very useful auxiliary lemma.

Lemma 3 *Let $G = (V, E)$ be a δ-hyperbolic graph. For every vertices $c, v, x, y \in V$, $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ or $d(y, v) - d(x, y) \geq d(c, v) - d(x, c) - 2\delta$ holds.*

Proof. Assume, without loss of generality, that $d(x, c) + d(v, y) \leq d(y, c) + d(x, v)$. If also $d(c, v) + d(x, y) \geq d(y, c) + d(x, v)$ then, by δ-hyperbolicity of G, $d(c, v) + d(x, y) - d(y, c) - d(x, v) \leq 2\delta$, i.e., $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$. If $d(c, v) + d(x, y) \leq d(y, c) + d(x, v)$, then $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) \geq d(c, v) - d(y, c) - 2\delta$. ∎

As easy corollaries we get two useful results known also from [7]. Denote by $F(s) := \{v \in V : d(s, v) = e(s)\}$ the set of vertices furthest from s.

Corollary 2 *For every δ-hyperbolic graph G,* $\mathrm{diam}(G) \geq 2\,\mathrm{rad}(G) - 4\delta - 1$.

Proof. Let x, y be vertices of G such that $d(x,y) = \mathrm{diam}(G)$. Let c be a middle vertex of any shortest path connecting x with y. Apply Lemma 3 to c, v, x, y, where $v \in F(c)$. Without loss of generality, assume that $d(x,v) - d(x,y) \geq d(c,v) - d(y,c) - 2\delta$ holds. Then, since $d(x,y) \geq d(x,v)$, $d(y,c) \geq d(c,v) - 2\delta$. Hence, $d(x,y) = d(x,c) + d(c,y) \geq 2d(c,y) - 1 \geq 2d(c,v) - 4\delta - 1 = 2e(c) - 4\delta - 1 \geq 2\,\mathrm{rad}(G) - 4\delta - 1$. ∎

Corollary 3 *Let $G = (V,E)$ be a δ-hyperbolic graph. For every vertices $c, v \in V$ such that $v \in F(c)$, $e(v) \geq \mathrm{diam}(G) - 2\delta \geq 2\,\mathrm{rad}(G) - 6\delta - 1$.*

Proof. Apply Lemma 3 to c, v and vertices x, y such that $d(x,y) = \mathrm{diam}(G)$. Without loss of generality, assume that $d(x,v) - d(x,y) \geq d(c,v) - d(y,c) - 2\delta$ holds. Then, since $d(c,v) \geq d(c,y)$, $e(v) \geq d(x,v) \geq d(x,y) + d(c,v) - d(y,c) - 2\delta \geq d(x,y) - 2\delta = \mathrm{diam}(G) - 2\delta$. ∎

We are ready to analyze Algorithm 1. Let u_i and $a_i \in F(u_i)$ be the vertices picked in iteration i of the do-while loop. Let $K_i := \{u_1, u_2, \ldots, u_i\}$ and $L_i := \{a_1, a_2, \ldots, a_i\}$. According to the algorithm, u_1 is picket arbitrarily (as initially $L = \emptyset$), a_1 is the vertex furthest from u_1, $u_2 = a_1$ (as $L_1 = \{a_1\}$ is a singleton), a_2 is a vertex most distant from $u_2 = a_1$, u_3 is a middle vertex of a shortest (a_1, a_2)-path. By Corollary 3, we already have $d(a_1, a_2) \geq \mathrm{diam}(G) - 2\delta$. We can also show that $e(u_3) \leq \mathrm{rad}(G) + 3\delta$.

Proposition 4 *If G is a δ-hyperbolic graph, then $d(a_1, a_2) \geq \mathrm{diam}(G) - 2\delta$ and $e(u_3) \leq \mathrm{rad}(G) + 3\delta$.*

Proof. We need only to estimate the eccentricity of the vertex u_3. As u_3 is a middle vertex of a shortest (a_1, a_2)-path, $\min\{d(c,a_1), d(c,a_2)\} \geq \lfloor \frac{d(a_1,a_2)}{2} \rfloor \geq \lfloor \frac{\mathrm{diam}(G)}{2} \rfloor - \delta$. Now, without loss of generality, assume (see Lemma 3) that for vertices $u_3, a_3, a_2, a_1 \in V$, $d(a_2, a_3) - d(a_2, a_1) \geq d(u_3, a_3) - d(a_1, u_3) - 2\delta$ holds. Then, $e(u_3) = d(u_3, a_3) \leq d(a_2, a_3) - d(a_2, a_1) + d(a_1, u_3) + 2\delta = d(a_2, a_3) - (a_2, u_3) + 2\delta \leq \mathrm{diam}(G) - \lfloor \frac{\mathrm{diam}(G)}{2} \rfloor + \delta + 2\delta = \lceil \frac{\mathrm{diam}(G)}{2} \rceil + 3\delta \leq \lceil \frac{2\,\mathrm{rad}(G)}{2} \rceil + 3\delta = \mathrm{rad}(G) + 3\delta$. ∎

Thus, in δ-hyperbolic graphs, a vertex with eccentricity at most $\mathrm{rad}(G) + 3\delta$ and a pair of vertices that are at least $\mathrm{diam}(G) - 2\delta$ apart from each other are found by Algorithm 1 in at most 3 iterations, i.e., in linear time. Note that a similar linear time algorithm was reported already in [7]: in δ-hyperbolic graphs, a vertex with eccentricity at most $\mathrm{rad}(G) + 5\delta$ and a pair of vertices that are at least $\mathrm{diam}(G) - 2\delta$ apart from each other can be found in linear time.

Next, we show that a vertex with eccentricity at most $\mathrm{rad}(G) + 2\delta$ is found by Algorithm 1 in at most $2\delta + 2$ iterations. Consider iteration $i \geq 3$ and let p', p'' be vertices of L_{i-1} with the largest distance, i.e., $d(p', p'') = \max\{d(x,y) : x, y \in L_{i-1}\} =: \mathit{diam}(L_{i-1})$. If $e(u_i) > \mathrm{rad}(G) + 2\delta$ then, applying Lemma 3 to $u_i, a_i, p', p'' \in V$, we get $d(p', a_i) - d(p', p'') \geq d(u_i, a_i) - d(p'', u_i) - 2\delta$ or $d(p'', a_i) - d(p', p'') \geq d(u_i, a_i) - d(p', u_i) - 2\delta$. Hence, $\max\{d(p', a_i) - d(p', p''), d(p'', a_i) - d(p', p'')\} \geq e(u_i) - \mathrm{rad}(G) - 2\delta > 0$ (as $\max\{d(p'', u_i), d(p', u_i)\} \leq \min_{u \in V} e_{L_{i-1}}(u) \leq \mathrm{rad}(G)$). That is, if $e(u_i) > \mathrm{rad}(G) + 2\delta$ then $\mathrm{diam}(L_i) > \mathrm{diam}(L_{i-1})$. As $\mathrm{diam}(L_2) = d(a_1, a_2) \geq \mathrm{diam}(G) - 2\delta$, in at most $2\delta + 2$ iterations of the while-loop of Algorithm 1 we will get $\mathit{diam}(L_i) = \mathit{diam}(L_{i-1})$ (with $i \leq 2\delta + 2$) and hence $e(u_i) \leq \mathrm{rad}(G) + 2\delta$ must hold. Thus, we proved the following proposition.

Proposition 5 *If G is a δ-hyperbolic graph, then there is an index $i \leq 2\delta + 2$ such that $e(u_i) \leq \mathrm{rad}(G) + 2\delta$. Furthermore, $e(u_j) \leq \mathrm{rad}(G) + 2\delta$ for all $j \geq 2\delta + 2$.*

The second part of Proposition 5 says that all u_i vertices generated by Algorithm 1 after $2\delta + 1$ iterations have eccentricity at most $\mathrm{rad}(G) + 2\delta$. Hence, in δ-hyperbolic graphs where the set $C^{2\delta}(G) := \{c \in V : e(c) \leq \mathrm{rad}(G) + 2\delta\}$ has cardinality bounded by some function $g(\delta)$, depending only on δ, our algorithm will produce a vertex with eccentricity $\mathrm{rad}(G)$ (i.e., a central vertex) in at most $g(\delta) + 2\delta + 1$ iterations.

Next we show that the set $C^{2\delta}(G)$ of a δ-hyperbolic graph has bounded diameter. Earlier, it was known that $diam(C(G)) \leq 4\delta + 1$ and there exists a vertex $c \in V$ such that $d(v, c) \leq 5\delta + 1$ for every $v \in C(G)$ [7].

Proposition 6 *If G is a δ-hyperbolic graph, then for every $x, y \in C^{2\delta}(G)$, $d(x, y) \leq 8\delta + 1$. Furthermore, there is a vertex $c \in V$ such that $d(v, c) \leq 6\delta$ for every $v \in C^{2\delta}(G)$.*

Proof. Let c be a middle vertex of any shortest path connecting x with y. Apply Lemma 3 to c, v, x, y, where $v \in F(c)$. Without loss of generality, assume that $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ holds. Then, $d(x, c) = d(x, y) - d(y, c) \leq d(x, v) - d(c, v) + 2\delta \leq e(x) - e(c) + 2\delta \leq \mathrm{rad}(G) + 2\delta - \mathrm{rad}(G) + 2\delta = 4\delta$. Hence, $d(x, y) = d(x, c) + d(y, c) \leq 2d(x, c) + 1 \leq 8\delta + 1$.

To prove the second assertion, consider a pair of vertices $x, y \in V$ with $d(x, y) = \mathrm{diam}(G)$ and a middle vertex c of any shortest (x, y)-path. Apply Lemma 3 to c, v, x, y, where v is an arbitrary vertex from $C^{2\delta}(G)$. Without loss of generality, assume that $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ holds. We know also that $d(x, c) \geq \lfloor \frac{d(x,y)}{2} \rfloor = \lfloor \frac{\mathrm{diam}(G)}{2} \rfloor \geq \lfloor \frac{2\,\mathrm{rad}(G) - 4\delta - 1}{2} \rfloor = \mathrm{rad}(G) - 2\delta$ (see Corollary 2). Hence, $d(c, v) \leq d(x, v) - d(x, y) + d(y, c) + 2\delta = d(x, v) - d(x, c) + 2\delta \leq e(v) - \mathrm{rad}(G) + 2\delta + 2\delta \leq \mathrm{rad}(G) + 2\delta - \mathrm{rad}(G) + 4\delta = 6\delta$. ∎

If the vertex degrees of a δ-hyperbolic graph are bounded by a constant Δ then $C^{2\delta}(G)$ has at most $\Delta^{O(\delta)}$ vertices. Summarizing, we have the following result.

Theorem 7 *Let $G = (V, E)$ be a δ-hyperbolic graph with m edges. Algorithm 1 finds*

1. *a vertex with eccentricity at most* $\mathrm{rad}(G) + 3\delta$ *in at most $O(m)$ time,*
2. *a vertex with eccentricity at most* $\mathrm{rad}(G) + 2\delta$ *in at most $O(\delta m)$ time,*
3. *a central vertex in at most $O(m)$ time, if the vertex degrees and δ are bounded by constants.*

Another linear time algorithm for finding a central vertex of a δ-hyperbolic graph with δ and vertex degrees bounded by constants was proposed in [7].

12 Chordal graphs

Recall that $F(s) = \{v \in V : d(v, s) = e(s)\}$ denotes the set of all vertices of G that are furthest from s and $C(G) = \{c \in V : e(c) = \mathrm{rad}(G)\}$ denotes the set of all central vertices of G. The *metric interval* $I(u, v)$ between vertices u and v is defined by $I(u, v) = \{x \in V : d(u, x) + d(x, v) = d(u, v)\}$, i.e., it consists of all vertices of G laying on shortest paths between u and v.

In this section we analyze the behavior of our algorithms in the class of chordal graphs. Recall that a graph G is *chordal* if every its induced cycle of length at least 4 has a chord. Chordal graphs are interesting because a central vertex in them can be found in linear time [12] but finding the diameter in linear time will refute the Orthogonal Vector Conjecture [13, 25].

First we give an example of an n-vertex chordal graph G on which Algorithm 1 will need $n/2$ iterations although G has a certificate for the radius consisting of only two vertices in L. Set $n = 2k$ and consider two sets of vertices $X = \{x_1, \ldots, x_k\}$ and $Y = \{y_1, \ldots, y_k\}$. The vertex set of G is $X \cup Y$. Make X a clique and Y an independent set in G. Make every vertex x_i adjacent to all vertices y_j with $j \leq i$. Algorithm 1 may place vertices $x_1, y_2, x_2, x_3, \ldots, x_k$ (in this order) into K and vertices $y_2, y_1, y_3, y_4, \ldots, y_k$ (in this order) into L. The central vertex x_k will be determined only when all Y-vertices are in L. On the other hand, $(x_k, \{y_1, y_k\})$ is a certificate for the radius of G.

Note that the graph G constructed has vertices of large degrees (up-to $n-1$). As every chordal graph G has the hyperbolicity at most 1, it follows from Theorem 7 that our algorithm finds a central vertex in linear time in every chordal graph with vertex degrees bounded by a constant. It should be noted that there is a linear time algorithm that finds a central vertex of an arbitrary chordal graph [12]; it uses additional metric properties of chordal graphs.

To analyze possible radius and diameter certificates in the class of chordal graphs, we will need the following important lemma.

Lemma 4 ([10]) *Let G be a chordal graph. Let x, y, v, u be vertices of G such that $v \in I(x, y)$, $x \in I(v, u)$, and x and v are adjacent. Then $d(u, y) \geq d(u, x) + d(v, y)$. Furthermore, $d(u, y) = d(u, x) + d(v, y)$ if and only if there exist a neighbor x' of x in $I(x, u)$, a neighbor v' of v in $I(v, y)$ and a vertex w with $N(w) \supseteq \{x', x, v, v'\}$; in particular, x', v' and w lay on a common shortest path of G between u and y.*

Our analysis is based on the following propositions which are also of independent interest. Recall that $C^1(G) := \{v \in V : e(v) \leq \mathrm{rad}(G) + 1\}$.

Proposition 7 *Let $G = (V, E)$ be a chordal graph.*

(i) If $\mathrm{diam}(G) < 2\,\mathrm{rad}(G)$ then, for every vertices $s \in V$ and $t \in F(s)$, there is a vertex $w \in I(s, t) \cap C(G)$ such that $t \in F(w)$.

(ii) If $\mathrm{diam}(G) = 2\,\mathrm{rad}(G)$ then, for every vertices $s \in V$ and $t \in F(s)$, there is a vertex $w \in I(s, t) \cap C^1(G)$ such that $t \in F(w)$.

Proof. First we show that for every vertex x of G with $e(x) = k > \mathrm{rad}(G)$ there is a vertex y such that $e(y) = k - 1$ and $d(x, y) \leq 2$. This is true even in the case when $\mathrm{diam}(G) = 2\,\mathrm{rad}(G)$. Consider a vertex y in G with $e(y) = k-1$ that is closest to x. Let z be any neighbor of y in $I(y, x)$. Necessarily, $e(z) = k$. Consider a vertex $u \in F(z)$. Since $d(y, u) \leq e(y) = k - 1 = e(z) - 1 = d(z, u) - 1 \leq d(y, u)$, we have $y \in I(z, u)$. Applying Lemma 4 to $y \in I(z, u)$ and $z \in I(y, x)$, we get $d(x, u) \geq d(x, y) - 1 + d(y, u) = d(x, y) + k - 2$. As $d(x, u) \leq e(x) = k$, we conclude $d(x, y) \leq 2$.

Next we claim that if $\mathrm{diam}(G) < 2\,\mathrm{rad}(G)$ then for every vertex x of G with $e(x) = k > \mathrm{rad}(G)$ there is in fact a vertex $z \in N(x)$ such that $e(z) = k - 1$. Furthermore, if $\mathrm{diam}(G) = 2\,\mathrm{rad}(G)$, such a neighbor z with $e(z) = k - 1$ exists for every vertex x of G with $e(x) = k > \mathrm{rad}(G) + 1$. Assume, by way of contradiction, that no neighbor of x has eccentricity $k - 1$ and let y be an arbitrary vertex of G with $d(x, y) = 2$ and $e(y) = k - 1$. Let also z be a vertex from $N(x) \cap N(y)$ for which the set $S_x(z) = \{v \in F(x) : z \in I(x, v)\}$ is largest. Necessarily, $e(z) = k$. As before, consider a vertex $u \in F(z)$. By Lemma 4, applied to $y \in I(z, u)$ and $z \in I(y, x)$, we get $d(x, u) \geq d(y, u) + d(x, z) = k - 1 + 1 = k$. As $d(x, u) \leq e(x) = k$, we conclude $d(x, u) = k$. Hence, by the second part of Lemma 4, there must exist a vertex w adjacent to y, z, x and at distance $k-1$ from u. As $u \in F(x)$, $w \in I(x, u), z \notin I(x, u)$, by the maximality of $|S_x(z)|$, there must exist a vertex $u' \in F(x)$ with $w \notin I(x, u'), z \in I(x, u')$. We have $d(z, u') = k - 1$, $d(w, u') = k$, and hence $z \in I(w, u')$ and $w \in I(z, u)$. By Lemma 4, $d(u, u') \geq d(u, w) + d(z, u') = k - 1 + k - 1 = 2k - 2$. Hence, $\mathrm{diam}(G) \geq d(u, u') > 2\,\mathrm{rad}(G)$, if $k > \mathrm{rad}(G) + 1$, and $\mathrm{diam}(G) \geq d(u, u') \geq 2\,\mathrm{rad}(G)$, if $k = \mathrm{rad}(G) + 1$. These contradictions prove the claim.

Now we can conclude our proof. Consider arbitrary vertices $s \in V$ and $t \in F(s)$ and proceed by the induction on $k = e(s)$. If $k = \mathrm{rad}(G)$ then $w = s$ and we are done. If $k = \mathrm{rad}(G) + 1$ and $\mathrm{diam}(G) = 2\,\mathrm{rad}(G)$ then again $w = s$ and we are done. If $k > \mathrm{rad}(G) + 1$ or $k = \mathrm{rad}(G) + 1$ and $\mathrm{diam}(G) < 2\,\mathrm{rad}(G)$ then a neighbor z of s with $e(z) = k - 1$ satisfies $t \in F(z)$, and we can apply the induction hypothesis. ∎

A pair x, y of vertices is called a *diametral pair* of a graph G if $d(x, y) = \mathrm{diam}(G)$.

Proposition 8 *The center $C(G)$ of a chordal graph G is a diameter certificate of G (not necessarily a smallest one).*

Proof. Additionally to Proposition 7*(i)*, we need to mention only that in any graph G with $\operatorname{diam}(G) = 2\operatorname{rad}(G)$, for every vertex $v \in C(G)$ and every diametral pair of vertices x, y, $d(x,y) = d(x,c) + d(y,c) = d(x,c) + \operatorname{rad}(G) = d(y,c) + \operatorname{rad}(G) = 2\operatorname{rad}(G)$ holds. ∎

Proposition 9 *For every chordal graph G, the set $C^1(G)$ is a tight upper certificate of G (not necessarily a smallest one).*

Proof. The statement follows from Proposition 7 and the definition of a tight upper certificate. ∎

So, it is interesting that if the center $C(G)$ is known for a chordal graph G then its diameter can be computed in $O(|C(G)||E|)$ time. However, there is no way to bound the cardinality of the set $C(G)$ in an arbitrary chordal graph G. In fact $C(G)$ may contain $n-2$ vertices in some chordal graphs. To construct such a graph G, take a complete graph K_{n-2} on $n-2$ vertices. Add two new vertices u and v adjacent to all vertices of K_{n-2} but not to each other. It is easy to see that $C(G)$ is exactly the vertices of K_{n-2}.

Nevertheless, it is known that for every chordal graph G and any two vertices x, y from $C(G)$, $d(x,y) \leq 3$ holds [11]. This suggest the following approach for computing the diameter of a chordal graph $G = (V,E)$.

- Use linear time algorithm from [12] to find a central vertex c of G.
- Set $C_3 := \{x \in V : d(x,c) \leq 3\}$. /* $C(G) \subseteq C_3$ */
- Find a vertex p such that $e^{C_3}(p)$ is maximum.
- Report $e(p)$.

The complexity of this approach is $O(|C_3||E|)$. As a consequence, we have that when the vertex degrees are bounded in a chordal graph by a constant Δ then its diameter can be computed in linear time (as $|C_3|$ is bounded by a constant Δ^3). We are not aware if such a result was known before. Note also that in general chordal graphs the cardinality of C_3 cannot be bounded by a constant since then the diameter of an arbitrary chordal graph can be computer in linear time refuting the Orthogonal Vector Conjecture [13, 25].

From the proof of Proposition 7 it follows also that, for every vertex v with $e(v) = \operatorname{rad}(G) + 1$, $d(v, C(G)) \leq 2$ holds. This suggest the following approach for computing the eccentricities of all vertices of a chordal graph $G = (V,E)$.

- Use linear time algorithm from [12] to find a central vertex c of G.
- Set $C_5 := \{x \in V : d(x,c) \leq 5\}$. /* $C^1(G) \subseteq C_5$ */
- For every vertex $v \in V$ report $e(v) = \min_{c \in C_5} d(v,c) + e(c)$.

The complexity of this approach is $O(|C_5||E|)$. As a consequence, we have that when the vertex degrees are bounded in a chordal graph by a constant Δ then the eccentricities of all its vertices can be computed in linear time (as $|C_5|$ is bounded by a constant Δ^5). We are not aware if such a result was known before.

Summarizing, we have the following result.

Theorem 8 *Let $G = (V,E)$ be a chordal graph with m edges and whose vertex degrees are bounded by a constant. Then, eccentricities of all vertices of G can be computed in total $O(m)$ time.*

By Proposition 8, the center $C(G)$ of a chordal graph G is a diameter certificate of G. Next we will show that the set of all diametral vertices of a chordal graph G forms a radius certificate of G.

Let $D^k(G) := \{v \in V : e(v) \geq diam(G) - k\}$. It is known that for every vertex v of a chordal graph G there is a vertex $u \in F(v)$ with $e(v) \geq diam(G) - 1$ [17]. Hence, the set $D^1(G)$ contains the output set L of Algorithm 1 and, therefore, gives already a radius certificate for a chordal graph G. In fact, we can prove a stronger result.

Proposition 10 *For an arbitrary graph G with $diam(G) \geq 2rad(G) - 1$, any diametral pair of vertices x, y forms a minimum radius certificate of G. Futhermore, $rad(G) = \lfloor \frac{d(x,y)+1}{2} \rfloor$.*

Proof. Let x, y be an arbitrary diametral pair of G, e.g., $d(x,y) = diam(G)$. If there is a vertex $z \in V$ with $\max\{d(z,x), d(z,y)\} \leq rad(G) - 1$ then $diam(G) = d(x,y) \leq d(z,x) + d(z,y) \leq 2rad(G) - 2$, and a contradiction arises. ∎

Proposition 11 *For every chordal graph G, the set $D(G) := D^0(G)$ is a radius certificate of G (not necessarily a smallest one).*

Proof. By Proposition 10 and the fact that in any chordal graph G, $diam(G) \geq 2rad(G) - 2$ [11], we need to consider only the case when $diam(G) = 2rad(G) - 2$.

Assume that there is a vertex u in G such that $d(u,t) \leq rad(G) - 1$ for every vertex $t \in D(G)$. Denote by S the set of all such vertices u. Denote by S' those vertices from S that have the minimum eccentricity. Finally, denote by S'' those vertices u from S' that have the smallest number of vertices in $F(u)$.

Consider a vertex $u \in S''$, a vertex $v \in F(u)$ and a neighbor w of u on a shortest path from u to v. Consider also an arbitrary vertex $x \in D(G)$ and an arbitrary vertex $y \in F(x) \subset D(G)$. Since $d(x,y) = 2rad(G) - 2$, we have $d(x,u) = d(y,u) = rad(G) - 1$ and hence $u \in I(x,y)$. We claim that $d(w,x) \leq rad(G) - 1$ as well. Assume that $d(w,x) > rad(G) - 1$. Then, $u \in I(x,w)$ and $w \in I(u,v)$. By Lemma 4, $d(x,v) \geq d(x,u) + d(w,v) \geq rad(G) - 1 + e(u) - 1 \geq 2rad(G) - 2$, i.e., $d(x,v) = 2rad(G) - 2$ (hence v must belong to $D(G)$) and $d(u,v) = e(u) = rad(G)$. The latter contradicts with the choice of u (as $u \in S$). Thus, $d(w,x) \leq rad(G) - 1$ for every vertex $x \in D(G)$, i.e., $w \in S$.

As $u \in S'$, $e(u) \leq e(w)$. First assume that $e(u) = e(w)$, i.e., $w \in S'$. Since $v \in F(u)$ and $v \notin F(w)$ (note that $d(w,v) = d(u,v) - 1 = e(u) - 1 \leq e(w) - 1$), by the choice of u (as $u \in S''$), there must exist a vertex $t \in V$ such that $t \in F(w)$ and $t \notin F(u)$. We necessarily have $d(t,u) = e(w) - 1 \geq rad(G) - 1$ and $d(v,w) = e(u) - 1 \geq rad(G) - 1$. One can apply now Lemma 4 to $u \in I(w,t)$ and $w \in I(u,v)$ and get $d(v,t) \geq d(v,w) + d(t,u) \geq 2rad(G) - 2 = diam(G)$. That is, both v and t must be in $D(G)$, contradicting again with the choice of u (as $u \in S$ and $d(u,v) = e(u) \geq rad(G)$).

Assume now that $e(u) < e(w)$ and consider an arbitrary vertex $t \in F(w)$. Necessarily, $u \in I(w,t)$. Hence again one can apply Lemma 4 to $u \in I(w,t)$ and $w \in I(u,v)$ and get $d(v,t) \geq d(v,w) + d(t,u) = e(u) - 1 + e(w) - 1 \geq 2rad(G) - 1 > diam(G)$, and a contradiction arises.

Contradictions obtained prove that for every vertex $u \in V$ there is a vertex $t \in D(G)$ such that $d(u,t) \geq rad(G)$, i.e., $D(G)$ is a radius certificate of G. ∎

13 Conclusion

A rough description for radius and diameter algorithms could be: take a node u not yet covered by a certificate C and add a node x to C such that u is now covered and possibly many other nodes are also covered. Selecting a node x covering u which is at maximum distance from u tends to provide a maximal covering set in both radius and diameter cases. Although this is not the classical greedy set-cover heuristic it is not surprising after all that it works well in practice.

References

[1] Amir Abboud, Fabrizio Grandoni, Virginia Vassilevska Williams: Subcubic Equivalences Between Graph Centrality Problems, APSP and Diameter. SODA 2015, pp..1681–1697.

[2] A. Abboud, J. Wang, V. Vassilevska Williams, Approximation and Fixed Parameter Subquadratic Algorithms for Radius and Diameter in Sparse Graphs, In Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, pp.. 377–391, Arlington, VA, USA, January 10-12, 2016.

[3] L. Backstrom, P. Boldi, M. Rosa, J. Ugander, S. Vigna: Four degrees of separation. WebSci, pp..33–42, 2012.

[4] M. Borassi, P. Crescenzi, M. Habib, Into the square - on the complexity of quadratic-time solvable problems, Electr. Notes Theor. Comput. Sci. 322: 51-67, 2016.

[5] Michele Borassi, Pierluigi Crescenzi, Michel Habib, Walter A. Kosters, Andrea Marino, Frank W. Takes: Fast diameter and radius BFS-based computation in (weakly connected) real-world graphs: With an application to the six degrees of separation games. Theor. Comput. Sci. 586: 59-80 (2015).

[6] Michele Borassi, Pierluigi Crescenzi, Luca Trevisan: An Axiomatic and an Average-Case Analysis of Algorithms and Heuristics for Metric Properties of Graphs. SODA 2017: 920-939.

[7] V. Chepoi, F. F. Dragan, B. Estellon, M. Habib, and Y. Vaxès, Diameters, centers, and approximating trees of δ-hyperbolic geodesic spaces and graphs, Symposuium on Computational Geometry (SoCG'08), ACM, New York, 2008, pp. 59–68.

[8] Pierluigi Crescenzi, Roberto Grossi, Michel Habib, Leonardo Lanzi, Andrea Marino, On computing the diameter of real-world undirected graphs, Theoret. Comput. Sci. 514 (2013) 84–95.

[9] Victor Chepoi, Bertrand Estellon: Packing and Covering delta-Hyperbolic Spaces by Balls. APPROX-RANDOM 2007: 59-73.

[10] V. Chepoi, Some d-convexity properties in triangulated graphs, in *Mathematical Research*, vol. 87, Ştiinţa, Chişinău, 1986, pp. 164–177 (Russian).

[11] V. Chepoi, Centers of triangulated graphs, *Math. Notes* 43 (1988), 143–151.

[12] V.D. Chepoi and F.F. Dragan, A Linear-Time Algorithm for Finding a Central Vertex of a Chordal Graph, ESA 1994: 159-170.

[13] D.G. Corneil, F.F. Dragan, M. Habib, C. Paul, Diameter determination on restricted graph families, *Discrete Applied Mathematics* 113(2001), 143–166.

[14] Derek G. Corneil, Feodor F. Dragan and Ekkehard Köhler, On the power of BFS to determine a graph's diameter, Networks vol. 42, 4, pp. 209–222, 2003.

[15] Derek G. Corneil, Jérémie Dusart, Michel Habib, Antoine Mamcarz, Fabien de Montgolfier, A tie-break model for graph search. Discrete Applied Mathematics 199: 89-100 (2016).

[16] Irit Dinur and David Steurer, Analytical Approach to Parallel Repetition, Symposium on Theory of Computing (STOC'14), ACM, 2014, pp. 624–633.

[17] Feodor F. Dragan, Falk Nicolai, Andreas Brandstädt: LexBFS-Orderings and Power of Graphs. WG 1996: 166-180

[18] M. Gromov, Hyperbolic groups, Essays in group theory, Math. Sci. Res. Inst. Publ., vol. 8, Springer, New York, 1987, pp. 75–263.

[19] P. Hage and F. Harary. Eccentricity and centrality in networks. Social Networks 17, pp. 57–63, 1995.

[20] G. Y. Handler, Minimax Location of a Facility in an Undirected Tree Graph, Transportation Science 7, pp. 287–293, 1973.

[21] Camille Jordan, Sur les assemblages de lignes, Journal für reine und angewandte Mathematik 70, pp. 185–190, 1869.

[22] Jure Leskovec and Andrej Krevl, SNAP Datasets: Stanford Large Network Dataset Collection, `http://snap.stanford.edu/data`, June 2014.

[23] Clémence Magnien, Matthieu Latapy, Michel Habib, Fast computation of empirically tight bounds for the diameter of massive graphs, ACM J. Exp. Algorithmics 13 (2009) 1–7.

[24] Mihai Patrascu, Ryan Williams: On the Possibility of Faster SAT Algorithms. SODA 2010: 1065-1075.

[25] L. Roditty, V. Vassilevska Williams, Fast approximation algorithms for the diameter and radius of sparse graphs, In Proceedings of the 45th annual ACM symposium on Symposium on theory of computing, STOC'13, pp..515–524, New York, NY, USA, 2013. ACM.

[26] V. Vassilevska Williams, Hardness of Easy Problems: Basing Hardness on Popular Conjectures such as the Strong Exponential Time Hypothesis, In Proceedings of the 10th International Symposium on Parameterized and Exact Computation (IPEC 2015), pp. 17–29, September 16-18, 2015, Patras, Greece, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik 2015.

[27] Frank W. Takes, Walter A. Kosters, Determining the diameter of small world networks, in: Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM, 2011, pp. 1191–1196.

[28] Frank W. Takes, Walter A. Kosters, Computing the eccentricity distribution of large graphs, Algorithms 6 (2013) 100–118.